IMPROVING CONTOURING ACCURACY IN CNC MACHINES

Xi Xuecheng
(B.Eng, M.Eng, NUAA, M.Eng, NUS)

A THESIS SUBMITTED
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
DEPARTMENT OF MECHANICAL ENGINEERING
NATIONAL UNIVERSITY OF SINGAPORE
JUNE 2008

Acknowledgements

First and foremost, I sincerely thank Prof. Poo Aun-Neow and Assoc Prof. Hong Geok-Soon, my supervisors, for their enthusiastic and continuous support and guidance. Their suggestion and ideas and critical comments have been crucial for the progress of this PhD project. During my PhD studies, they provided me not only with the technical guidance, but also strong encouragement and kind affection.

I thank also Mr. Mok Heng Chong for his valuable help in setting up the Mini-CNC and made it a running machine. I also thank Mr. Sakthi, Ms Ooi, Miss Tshin and many others in the Control and Mechatronics Lab for their help in the experiments.

I am grateful to Dr. Duan Kaibo, Mr. Wang Jie, Miss Yang Lin, Mr. Dau Van Huan, Miss Leong Ching Ying Florence, Miss Ghotbi Bahareh and many other friends for their invaluable friendship, advice and help during the project. Without their help and encouragement, I would not have carried out this study smoothly.

Finally, I thank my dear parents and wife for their unwavering support and encouragement. I thank my son for the joy that he brings to me. Their love gives me the power to move forward.

Table of Contents

Summary

With the increasing demand on the dimensional accuracy of machined parts, contouring accuracy in terms of contour error has been, and continues to be, a big concern in the design and control of continuous-path CNC machines. Current methods for achieving greater accuracies can be classified as control approaches or compensation approaches.

Several approaches are explored and developed in this thesis to improve the contouring accuracy of CNC machines. A straightforward approach is first investigated which to keep the dynamics of the machine simple with the use of a simple proportional controller for the position feedback loop. It is shown that with perfectly matched axial dynamics, perfect linear paths with no contour errors can be achieved. With the addition of a simple feedforward gain to compensate for radial errors resulting from limited bandwidth of the machine axes, perfect circular contours with no contour errors can also be achieved. A tuning procedure, using measured steady-state axial tracking errors, is then proposed to tune the gains so as to achieve matched axial dynamics. This approach is experimentally shown to work well and, on a target mini-CNC machine, was able to reduce contour errors, for both linear and circular paths in the steady-state, to within just a few feedback resolution or basic length unit. The remaining significant contour errors are then those caused by stiction when starting from standstill or at velocity reversals.

A two-stage stiction compensation scheme is proposed to reduce or eliminate the contour errors caused by stiction. Experimental investigations show that this compensation method is effective in reducing the error spikes at the quadrant positions in circular contours.

A model-based Taylor series expansion error compensation (TSEEC) approach which

formulates the contour error as a Taylor series expansion around points along the desired path and compute compensation components as deviation from these points, is also proposed, developed and evaluated. With perfect knowledge of the machne's dynamics, simulation shows that TSEEC can achieve perfect contouring with zero contour errors for both linear and circular contours. Experiments carried out, using a dynamic model of the machine identified experimentally, also show very good contouring performance.

Finally, an integral sliding mode control (ISMC) approach, due to its robustness against model uncertainties and external disturbances, is developed and evaluated for reducing axal tracking errors. The step-by-step approach is used in the design of ISMC. Experimental results show that the ISMC can improve the contouring accuracy greatly, even at the quadrant positions where stiction occurs at the reversal of velocities and when starting from standstill.

List of Tables

List of Figures

List of Symbols

α	compensation gain
β	feedforward compensation coefficient
$\Delta r_x, \Delta r_y$	correction to reference input of X, Y axis
γ	ratio of e_x and e_y
λ	coefficient of mismatch
ω	angular velocity
ω_n	natural frequency of second order system
ϕ	phase angle of frequency response
τ	time constant of axial dynamics
θ	inclination angle with respect to X-axis
ε	contour error without compensation
ε^*	contour error with compensation
ζ	damping ratio of second order system
$A(q^{-1})$	denominator of closed-loop transfer function
A, B	partial derivative of contour error ε with respect to r_x, r_y
$B(q^{-1})$	numerator of closed-loop transfer function
$B^+(q^{-1})$	polynomial with acceptable zeros

$B^-(q^{-1})$ polynomial with unacceptable zeros

d_b breakaway displacement

e_x, e_y tracking error of X, Y axis

f feedrate

F_c compensation signal

K gain of axial dynamics

K_f feedforward gain

K_p proportional gain in position loop

k_f compensation coefficient

K_m proportional gain of a motor

M magnitude ratio of frequency response

P, P^* actual and desired positions

R radius

r_x^*, r_y^* reference (uncompensated) input to X, Y axis

r_x, r_y actual (compensated) input to X, Y axis

r_{fx}, r_{fy} filtered reference input to X, Y axis

T sampling interval

u_{max} maximum compensation value

V path velocity

v^* dimensionless velocity

v_b breakaway velocity

v_r reference velocity

Chapter 1

Introduction

Computer numerical control (CNC) machine tools are now widely used in the manufacturing industry. With an increasing demand on the dimensional accuracy of machined parts, researchers are continuing to seek various methods to improve the machining accuracy of CNC machines. Contouring accuracy in terms of contour error is a big concern for the designers and end-users of contouring (or continuous-path) type of CNC machines. Contour error is defined as the orthogonal component of the deviation of the actual contour from that desired.

In CNC machines, a part is manufactured by a part program which defines the geometrical dimensions and manufacturing conditions such as feedrate and tool type. The part program can be manually written or produced by a computer-aided manufacturing (CAM) program.

One class of CNC systems is the contouring systems, in which the tool is cutting while the axes of motion are moving. An example of this is the CNC milling machine. In these systems, the machine axes are separately driven and controlled so that they follow the reference inputs generated by an interpolator. The interpolator coordinates

the motion among different axes by supplying the corresponding reference inputs to each axis of motion so as to generate tool paths necessary to machine the desired part. In most modern CNC machines, the interpolator is capable of interpolating linear, circular and occasionally parabolic contours.

The reference motion commands generated by the interpolator are sent to the position feedback control loops of the machine. While the reference trajectories generated by the interpolator only define the ideal tool path for the machine axes to follow, the actual tool path or trajectories will deviate from that desired due to various causes including [43, 44, 45, 47, 46]:

1. Geometrical inaccuracies of the machine structure. These could be due to misalignment of the machine axes, inaccuracies and lack of straightness in machine slides, or runout errors (the radial variations from a true circle) of rotating machines [44],

2. Inaccuracies of position transducers used in the position feedback control loop,

3. Deflections and vibrations. Inaccuracies can arise due to deflection in the machine structure either due to shifting of weights as the worktable or tool moves or due to forces generated during machining. Unwanted vibrations and chatter can also be caused when resonant frequencies are excited by cutting forces [44],

4. Thermal expansion due to temperature changes either due to environment factors or to heat generated during cutting operation [45, 47], and

5. Errors caused by tracking or following errors in the position feedback control loops due to imperfect control [43, 46].

Static geometrical inaccuracies, including misalignment of machine axes and errors in the position transducers are commonly, and relatively easily, compensated for by calibration with an accurate measuring system such as a laser interferometer, and the use of a look-up table to store axes error compensation data.

Much work has been done, and ongoing to reduce errors due to thermal and dynamic effects. In this thesis only the *control contour errors*, or the contour errors resulting from imperfect control, or coordination of the motion, of the machine axes are investigated. Control contour errors can arise due to the existence of unmatched axial dynamics, nonlinearities and external disturbances.

Tracking error is the difference between the reference command and the actual position. *Contour error* at any point along a tool path is the orthogonal deviation of the actual tool trajectory from the desired trajectory at that point (i.e., the deviation of the cutter location from the desired path). In contouring systems, we are more concerned with contour errors rather than tracking errors or following errors of individual axes. It is quite possible that even when tracking errors are present in the individual axes, these can be made to cancel each other out so as to achieve zero contour error [43].

Efforts to reduce or eliminate control contour errors (hereafter in this thesis this will refer to control contour errors) achieved either through (1) the design of advanced controllers for feed drives, or by (2) applying compensation at the reference inputs.

Among the first category of advanced controller design, judging from the ways through which contour errors are eliminated or reduced, there are tracking control and contouring control. In tracking control, efforts are made to eliminate or reduce the axial

tracking errors and thereby reduce the contour errors indirectly. By contrast, in contouring control, the objective is to eliminate or reduce the contour errors directly.

The second main category of compensation schemes includes (1) iterative learning compensation, (2) path precompensation, (3) dynamic interpolation, and (4) compensation for nonlinearities.

In this thesis, several methods are proposed for improving the contouring accuracy in CNC machines. These include (1) an approach for axial dynamics matching through gain tuning, (2) a stiction compensation scheme to reduce the effects of stiction on contour errors, (3) a Taylor Series Expansion Error Compensation (TSEEC) scheme, and (4) an integral sliding mode control (ISMC) approach. Depending on their different nature, these methods belong to different categories in terms of the way they achieve improvements to contouring accuracy. More specifically, gain tuning belongs to contouring control. The stiction compensation schemes is a control approach to compensate for the effect of stiction. The TSEEC scheme employs feedforward compensation at the reference inputs and the integral sliding mode control belongs to the category of tracking control, aiming at reducing the tracking errors and thus indirectly reducing the contour errors.

A brief introduction to the rest of the chapters of this thesis follows.

Chapter 2 first gives a general introduction to CNC machines and a classification of these machines. This is followed by a comprehensive literature review of the two main categories of the ways of improving contouring accuracy in CNC machines, i.e., the design of advanced controllers for feed drives and compensation applied at the reference inputs.

Chapter 3 investigates some factors that affect the contouring accuracy by simulation. Here, it is shown that by making the dynamics of position control loops simple and matching axial dynamics through the matching of loop gains, very significant improvements to contouring accuracies can be achieved. For perfectly matched axial dynamics, perfect contouring can be achieved for linear contours and, together with feedforward radial compensation, also for circular contours.

Chapter 4 presents a simple and straightforward approach to match axial dynamics by tuning control loop gains. Experimental results show that the contouring accuracy for both linear and circular contours can be very significantly improved as a result, particularly average contour errors at steady state when stiction effects are not present. This chapter also presents a simple method of feedforward compensation at the reference inputs which can effectively eliminate, for circular contours, the radial errors resulting from the limited bandwidth of the axial dynamics.

Chapter 5 proposes a stiction compensation scheme to reduce the effects of stiction on contour errors at the quadrant positions in a circular contour. At these positions, the axes of motion experience a reversal of velocities or a start from standstill. This is a two-stage static friction compensation scheme in which the compensating signal is a continuous signal comprising an increasing portion followed by a decreasing portion. Experimental results show that this scheme can effectively reduce contour errors due to stiction to levels comparable to those caused by machine vibrations.

The aforementioned approaches are essential model-free approaches for which a knowlege of the dynamic model of the machine axes is not necessary. In Chapter 6 and 7, two model-based approaches for reducing contour errors are presented.

Chapter 6 presents a Taylor series expansion error compensation (TSEEC) scheme. In TSEEC, the contour error compensation problem is formulated as a Taylor series expansion problem, in which the value of the contour error is expanded around the reference points and the compensation components are calculated as the deviations from these reference points. It is shown, using simulation, that if perfect knowledge of the dynamic models of the machine axes is known, then perfect contouring with zero contour errors can be achieved using TSEEC. However, because of imperfect knowledge of the dynamic models, and to cater to other effects such as external disturbances and measurement noise, some modifications are necessary. Experimental results show that TSEEC can effectively reduce contour errors.

In Chapter 7, a tracking control strategy based on an integral sliding mode control (ISMC) approach is presented and discussed. ISMC is investigated here to reduce the tracking errors and improving contouring accuracy because of its known robustness against model uncertainties and external disturbances. The design of ISMC is a step-by-step approach with the first stage the design of a two-degree-of-freedom controller (or RST controller) with independent objectives for regulation and tracking, and the second stage an equivalent control which incorporates the estimate of the disturbances into the RST control law. The final stage is driving the system dynamics onto a predefined sliding surface with the integral of the switching function added to the control law. It is shown, experimentally, that by using ISMC for both axes, the contouring accuracy can be greatly improved, even at the quadrant positions where stiction occurs at the reversal of velocities and at the start from standstill.

The conclusions and lessons learned from this work are presented and discussed in Chapter 8 where the author's contributions are also highlighted. Some further topics

are also proposed for possible future research.

The Appendix gives a detailed description of how the dynamic models of the machine axes are identified using a closed-loop identification method. Two measures are taken to improve the quality of the obtained models. The first is to take into account the *a priori* knowledge of the existence of an integrator. The second is to scale the input-output data so that the magnitude of the input data are in the same range of that of the output data.

Chapter 2

Literature Review

In this chapter, a comprehensive literature review of general approaches to improve contouring accuracy in CNC machines is presented. Section 2.1 first proceeds with the general background and three classifications of CNC machines. In this thesis, the focus is on closed-loop, sampled-data and contouring CNC machines. Section 2.2 gives the definition of contour error and a summary of two categories of approaches to improve contouring accuracy. Section 2.3 reviews the methods of improving contouring accuracy by designing various advanced controllers, including feedback, feedforward and cross-coupled controllers. Section 2.4 introduces different kinds of compensation techniques for reducing contour errors.

2.1 An introduction to CNC machine tools

Controlling a machine tool by means of a prepared program which contains numerical data specifying the desired motion of the machine's axes is known as numerical control (NC). NC is the result of a research sponsored by the US Air Force in the

early 1950s, when there was an increasingly need for viable methods to manufacture complicated and accurate parts for aircrafts. The first NC milling machine was developed by the Servomechanism Laboratory in the Massachusetts Institute of Technology in 1952 [17]. Computer numerical control (CNC) is an extension to NC with the hardwired NC controller replaced by a software-driven computer. The first CNC system appeared in the early 1970s with the first minicomputers and microcomputers were developed. With the rapidly increasing capabilities and speeds of computer systems, these software-based CNC systems bring with them much greater flexibilities and capabilities. It is now quite possible to incorporate intelligence into the controllers of these CNC systems to improve both performance in terms of accuracies and productivity.

Rather than going directly to the contouring accuracy of CNC machines, which is the main topic of this thesis, some relevant aspects of CNC machines will first be introduced.

There are different ways in which CNC systems can be classified. According to the form of computer, or interpolator, output which, in turn, also determines the form of the position feedback loop employed, there are reference pulse (pulse or incremental output) systems and sampled-data (reference word output) systems. If based on the type of position control loops, there are the open-loop (stepper motor driven) and the closed-loop (servo motor driven) CNC systems. According to the type of machining needed, which is of significance to a manufacturing process, there are the point-to-point (or positioning) and the contouring (or continuous path) CNC systems. In this thesis, we will focus only on closed-loop, sampled-data and contouring CNC systems.

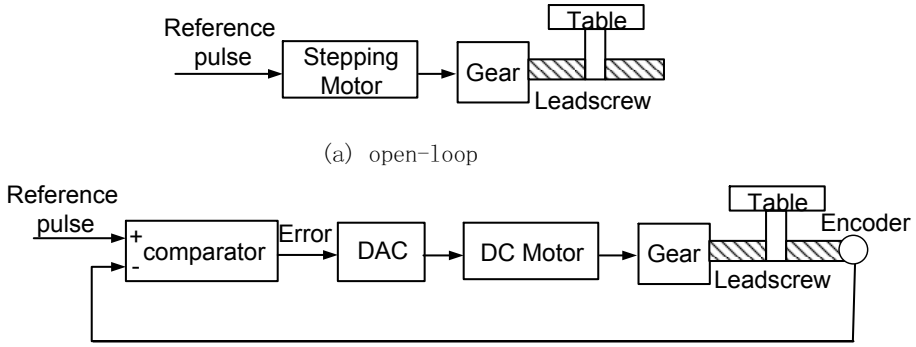

Figure 2.1: Reference-pulse systems: (a) open-loop and (b) closed-loop stepping system [28].

2.1.1 Interpolation and forms of computer output

For a contouring system, it is necessary to coordinate the movements of the separately driven axes of motion to achieve a desired path of the tool relative to the workpiece. This involves the generation of reference axes position commands based on the desired shape of the workpiece and their transmission as reference inputs to the axial position control loops. The generation of these reference commands is accomplished by interpolators [28]. Interpolators in typical modern-day CNC systems are capable of generating linear, circular, and occasionally parabolic paths.

Basically, pulse and binary word are the two forms of output that a computer can output to the external axial drives. Accordingly to this, we can classify CNC systems into two types: the reference-pulse and sampled-data systems. In reference pulse systems, the computer produces a sequence of reference pulses for each axis of motion. Each pulse, either in the positive or negative direction, represents an incremental motion of a basic length unit (BLU) in that direction. The accumulated number

of pulses represents the displacement, and the frequency at which the pulses are generated will be proportional to the axial velocity. The pulses can either actuate a stepping motor in an open-loop system as shown in Fig. 2.1 (a), or be fed as reference pulses to an external closed-loop position feedback system configured as a "stepping system" as shown in Fig. 2.1 (b) [28].

In sampled-data systems, the position feedback control loop of each axis is normally closed through the control computer itself. During each sampling period, the following processes will be performed:

- The interpolation routine generates a set of desired position references,
- The actual positions of each axis are sampled through the position feedback transducers,
- Using information on the desired reference positions, and the feedback positions, a digital controller generates the necessary control signal for each axis according to some desired control law, and
- The control signals are then sent to the axes drive systems through digital-to-analog (DAC) converters.

In sampled-data systems, the maximum feedrate is not limited by the interpolation frequency as compared with reference pulse systems. It is also possible to employ more advanced control algorithms in sampled-data systems while in reference pulse systems, only simpler control methods are implemented [28].

2.1.2 Control of axis of motion

The position control loops of CNC systems are designed to perform a specific task, and that is to control the position of the machine tool axes to accurately follow the reference position trajectories provided by the interpolator. Each axis of motion is separately driven by its own axial controller. According to the control configuration used, there are two types of controllers: open loop and closed loop. In the former type, stepping motors are used as drive devices, "stepping" or moving a constant incremental displacement for each output pulse from the interpolator. Stepping motors are usually used only for small-sized systems in which the torque requirement is small. Closed-loop controllers are usually required for improved accuracy and for large loads.

2.1.3 Point-to-point and contouring systems

According to the type of machining process required, CNC machines can either be classified as point-to-point (or positioning) systems or contouring (or continuous-path) systems. In point-to-point systems, the path of the machine tool when moving from the starting position to the end point is not important. What is of importance is only the accuracy of the positioning of the tool at the desired end point. In point-to-point systems, the tool is normally not in contact with the workpiece during motion. Example of such systems are CNC drilling machines or hole punching machines.

In contouring systems, on the other hand, the tool will be required to be cutting during motion and the accuracy of the "tool path" determines the contour accuracy

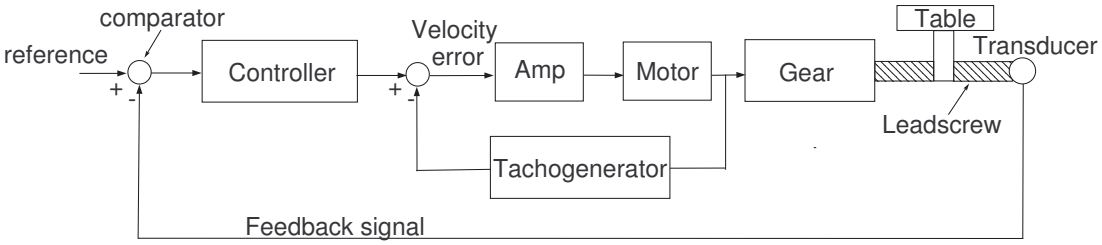

Figure 2.2: Control loop of contouring system [28].

of the machined part. Any deviation in the tool path causes an error in the shape of the part. A contouring system must therefore accurately control the path of the motion and not just the end positions. A typical closed-loop position control axis is shown in Fig. 2.2. Shown are two feedback devices: a tacho-generator for the inner velocity control loop and a position feedback transducer for the outer position feedback loop.

2.2 Contouring accuracy

The *contour error* is the deviation of the actual path taken by the machine's axes from the desired path. At any point on the path, it is defined by the orthogonal distance of the actual path from the desired path. The relationship between the contour error and the axial tracking, or following, errors on a bi-axial system is illustrated in Fig. 2.3. In contouring systems, contour errors are our primary concern rather than axial tracking errors.

In this figure, the desired path is shown as a solid line while the actual path taken is shown by the dashed line. The figure also illustrate an instance of time t when the desired position of the tool, as provided by the outputs of the interpolator, is at R

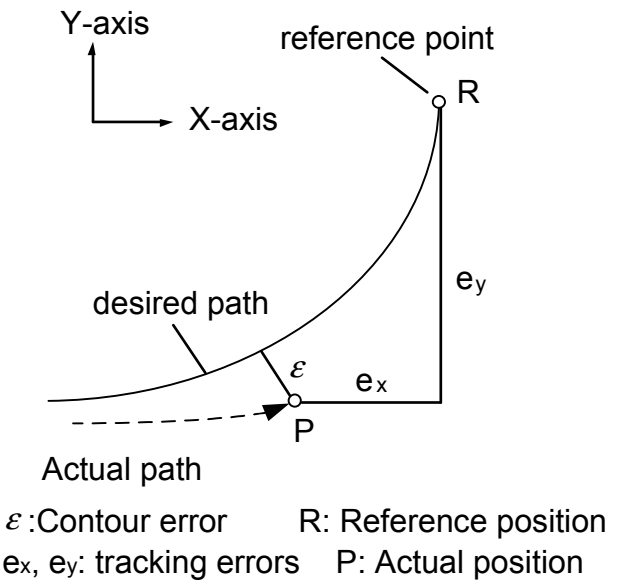

Figure 2.3: Contour error in machining a contour [30].

while, because of axial tracking errors, the actual position of the tool is at P. From the figure, it can be easily seen that, depending upon the relative magnitudes of the axial tracking errors, e_x and e_y, it is well possible to have zero contour errors even for non-zero axial tracking errors.

Efforts to reduce or eliminate contour errors have been made either through (1) the design of advanced controllers for feed drives, or by (2) compensation added to the reference inputs. For easy reference, the methods for improving contouring accuracy are listed in Table 2.1.

In terms of achieving contouring accuracies in CNC systems, controllers can be classified either as tracking control and contouring control. In tracking control, the primary objective is to reduce or eliminate the axial tracking errors (e_x and e_y in Fig. 2.3) and, thereby, reduce or eliminate the contour errors indirectly. By contrast, in contouring control, the objective is to eliminate or reduce contour errors directly (ε in

Table 2.1: Methods of improving contouring accuracy

Control	tracking control	feedforward feedback sliding mode control
	contouring control	cross-coupled feedback
Compensation	iterative learning path precompensation dynamic interpolation compensation for nonlinearities	

Fig. 2.3), whether or not tracking errors in individual axes are reduced. Based on the type of controllers, there are three basic approaches: (1) feedback controllers, (2) feedforward controllers, and (3) cross-coupled controllers [30, 46]. In tracking control, feedback, feedforward and sliding mode controllers have been used while for contouring control, feedback and cross-coupled controllers have been used. It can be noted from Table 2.1 that feedback controllers are used for both tracking as well as contouring control. Well designed feedback controllers are capable of reducing both tracking as well as contour errors in CNC machines.

The second main category for achieving better contouring accuracy is the use of error compensation which includes (1) iterative learning compensation, (2) path precompensation, (3) dynamic interpolation , and (4) compensation for nonlinearities.

2.3 Advanced controllers for feed drives

In this section, some advanced controllers used for feed drives to achieve better performance are presented and discussed.

2.3.1 Feedback controllers

It should be noted that the major function of feedback controllers is regulation against external disturbances and parameter variations. In the contouring control of CNC machines, the performance of feedback controllers can have significant effects on the contouring accuracy. One approach in the design of feedback controllers is under the framework of tracking control. For example, by tuning the feedback controller, a broader bandwidth of the axial dynamics can be achieved to reduce the tracking errors, thus indirectly reducing contour errors. The other way is under the framework of contouring control such that the dynamics of the different axes are matched.

Poo *et al.* [43] studied the effects of dynamic errors in two-axis type 1 contouring systems and started the work of analyzing relations between feedback controllers and contour errors. It was found that when the dynamics of two axes are matched, zero contour error can be achieved for a straight line trajectory even in the presence of significant tracking errors in the individual axes. In the case of circular contours with matched axial dynamics, a perfect circle is generated but the radius will be smaller or larger than the desired circle depending on the damping ratio and the angular velocity around the circular path. Mismatched gains usually would result in considerable contour errors. For good contouring accuracy, care should thus be made to ensure that the axial dynamics of CNC machines are properly matched.

Although many advanced control algorithms and structures have been developed, the

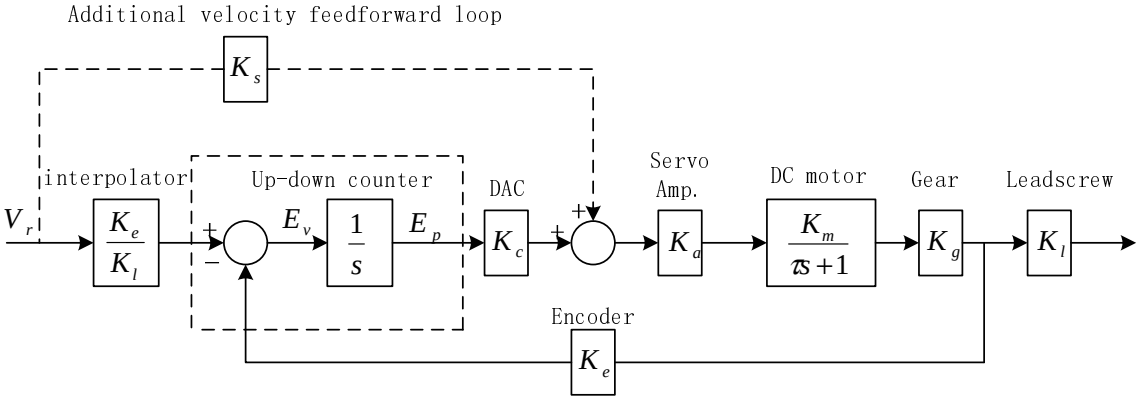

Figure 2.4: Additional velocity feedforward loop [36].

design of the feedback controller is still the most fundamental and crucial factor in obtaining desirable motion accuracy. To improve contouring accuracy in general multi-axis motion systems, feedback controllers should be designed to achieve matched dynamic characteristics among all axes. Yeh and Hsu [69] proposed a perfectly matched feedback control (PMFBC) design for multi-axis motion systems. By applying stable pole-zero cancelation and including complementary zeros for uncanceled zeros for all axes, matched dynamic responses across the whole frequency range for all axes were achieved. The performance of PMFBC is, however, highly dependant on the accuracy of the model, which is usually difficult to achieve in practice.

2.3.2 Feedforward controllers

Tracking controllers work to reduce or eliminate the tracking errors in the individual axial position feedback control loops, thereby indirectly reducing the contour errors. Masory [36] proposed that by adding a velocity feedforward loop to the conventional feedback controller as shown in Fig. 2.4, the position-following error can be reduced, or even eliminated, and consequently the contour error is reduced.

The basic idea is to close the servo loop through a feedback controller to take care of disturbances and parameter variations, and to cascade the closed-loop dynamics with a feedforward controller with a gain such that it cancels out most of its stable components, resulting in an overall transfer function equal to or very close to unity. If there are no unstable zeros of the plant, a suitable feedforward controller will simply be the inverse of the closed-loop plant, based on stable pole/zero cancelation, i.e. $G_0^{-1}(z)G_{close}(z) = 1$, where $G_0^{-1}(z)$ is the transfer function of the feedforward controller and $G_{close}(z)$ that of the closed-loop plant. If $G_0^{-1}(z)$ contains unstable poles, then this cannot be implemented as a feedforward controller, and an approximation will need to be used [30].

Suppose the closed-loop discrete-time transfer function, which includes the plant with the feedback controller, is expressed as:

$$G_{close}(z^{-1}) = \frac{z^{-d}B^{+}(z^{-1})B^{-}(z^{-1})}{A(z^{-1})} \tag{2.3.1}$$

where z^{-d} represents a delay of steps normally caused by the control loop, A includes the closed-loop poles, B^{+} includes the acceptable closed-loop zeros, and B^{-} includes the unacceptable closed-loop zeros. The "acceptable" zeros here mean the zeros that are located inside the unit circle and can be taken as the poles in the feedforward controller. By contrast, unacceptable zeros are located outside the unit circle and cannot be the poles of the feedforward controller. If unacceptable zeros exist, the feedforward controller cannot be implemented as the inverse of the plant. Tomizuka [56] modified the feedforward controller structure as shown in Fig. 2.5. This feedforward controller is referred to as the zero phase error tracking controller (ZPETC) which can achieve

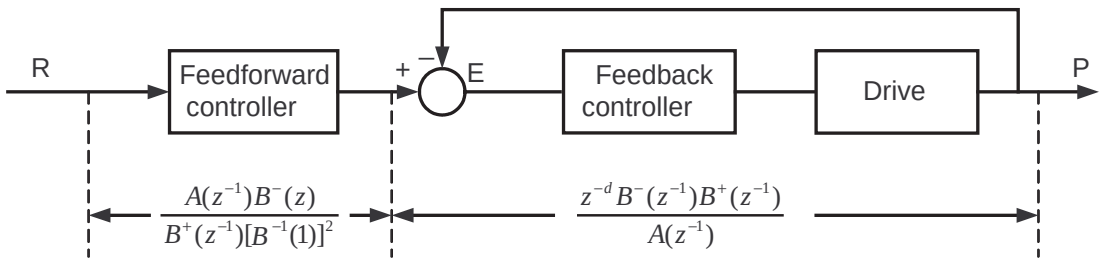

Figure 2.5: Zero phase error tracking control system [56].

zero phase lag and unity DC gain for any reference inputs. The original ZPETC performs well at low frequencies but tracking performance deteriorates as the frequency increases. Some variations of ZPETC have been proposed, which have been reported to achieve better performance [20, 58, 57].

2.3.3 Sliding mode control

In multi-axis applications, the straightforward approach to achieve good contouring accuracy is still through achieving good tracking performance. Sliding mode control (SMC) is well known for its robustness against model uncertainties and external disturbances. By driving the system dynamics onto a predefined sliding surface, the system dynamics are forced to behave in a desired way. For multi-axis applications, the same sliding surface can be designed for the different axes. Thus, on one hand, SMC can be used to reduce individual axial tracking errors, and on the other, match axial dynamics among the different axes.

Researchers utilized SMC to improve the contouring accuracy for CNC machines. Altintas *et al.* [2] proposed an adaptive sliding mode control approach for the control of high speed feed drives. It has been shown to be robust against uncertainties in

the drives' parameters, and be able to compensate for external disturbances such as friction and cutting force. Chen *et al.* [7] proposed two integral sliding mode controllers based on different characteristics of the model dynamics. Compared with conventional SMC, the chattering in the actuator was eliminated by the introduction of an integral action.

2.3.4 Cross-coupled controllers

While tracking control aims to improve the tracking accuracy of individual axes, the cross-coupled controller (CCC) proposed by Koren [27] is devised to reduce the contour errors directly, rather than achieve this by reducing the individual axial tracking errors.

The main idea of cross-coupled control is to build in real time a contour error model based on the feedback information from all axes as well as the interpolator to find an optimal compensating law, and then to feed back correction signals to the individual axes. A typical cross-coupled controller essentially consists of (1) an algorithm to calculate the contour error and (2) a control law to eliminate the contour error.

Later, variations to the original CCC were made. Koren and Lo [29] proposed a variable-gain cross-coupled controller in which the gains are adjusted in real-time according to the shape of the contour. The structure of their variable-gain CCC is shown in Fig. 2.6.

Srinivasan and Kulkarni [51] presented an approximate stability analysis of the cross-coupled controller and evaluated this experimentally on a microcomputer-controlled two-axis positioning table. The results were compared with those of a traditional

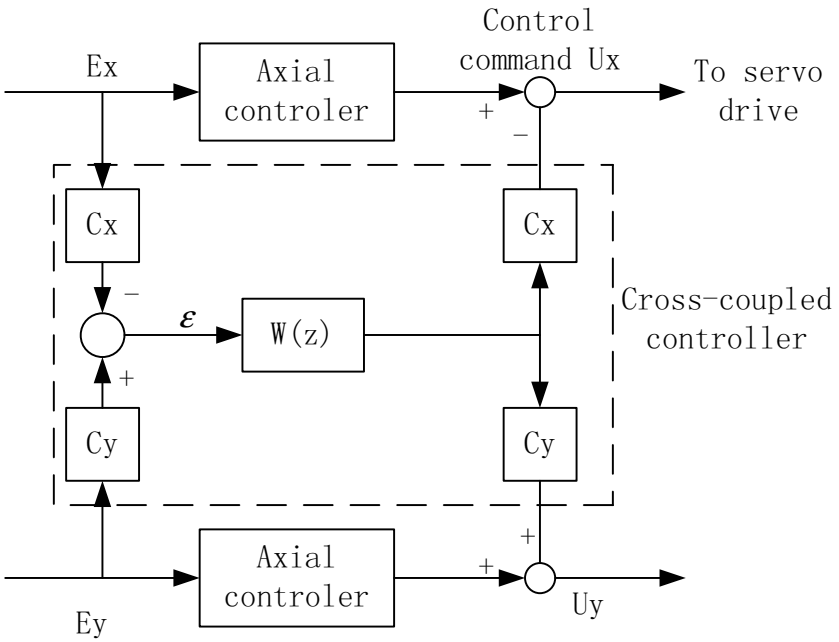

Figure 2.6: The variable-gain cross-coupled controller [30].

uncoupled controller.

Although cross-coupled controllers have been shown to be effective in reducing contour errors, increasing feedrates degrades the contouring accuracy. In order to improve the contouring accuracy for various types of contours with varying feedrates, Chuang and Liu [14] proposed a cross-coupled adaptive feedrate control strategy. To obtain feasible solutions, a linear perturbed model relating feedrate and contour error was established and self-tuning adaptive control laws were derived.

Chuang and Liu [15] further combined the model reference adaptive control strategy with the cross-coupled control of axial motion. By specifying a contour error bound, the desired feedrate is adaptively adjusted online so that the resultant contour errors can be maintained within the specified bound.

Zhong *et al.* [70] introduced a First-In-First-Out (FIFO) buffer into a cross-coupled controller to store the micro straight lines for curves generated by an interpolator. These micro straight lines were used to find the closest position vector P_1^* on the desired contour which is closest to the actual position vector P. The contour error E_r for an arbitrary curved contour can be calculated using $E_r = P_1^* - P$. By doing so, the cross-coupled controller becomes a linear controller which is easy for stability and steady-state analysis.

A cross-coupled control system is a multi-variable, nonlinear, and time-varying control system for which tuning the parameters is not a trivial problem. Some intelligent algorithms have been proposed to tune these parameters. Tarng *et al.* [54] first added a velocity feedforward controller to the velocity loop of the individual axes (Masory's method in [36]), and then used genetic algorithms to find the optimal parameters for the cross-coupled controller.

Yeh and Hsu [66] studied a robust CCC design based on a contour error transfer function (CETF), which resulted in an equivalent formulation as in the feedback control design problem. Methods in robust control design can then be readily employed to achieve robust CCC with specified stability margins and guaranteed contouring performance.

Combined with the multiple-loop cascaded control design method, Shih *et al.* [49] investigated a new CCC structure and analyzed its stability. While classical CCC modifies the commands to the velocity loop, the proposed structure allows CCC to directly compensate the reference commands to the position loop of the axes.

2.3.5 Coordinate transform

The idea of cross-coupled control is to directly reduce the contour errors rather than the tracking errors of individual axes. The traditional approach of designing a CCC controller in an X-Y coordinate system makes the analysis of stability and performance very difficult. A natural extension to existing contouring controllers is to design them in a coordinate system that makes the calculation of contour error and the design of the contouring controller easier. Several researchers have made efforts in this direction.

Lo and Chung [33] presented a controller in a tangential-contouring (T-C) frame for a biaxial system. The design of the controller is based on the coordinate transformation between the X-Y and the T-C coordinate bases.

Ho *et al.* [21] investigated a path-following control algorithm for a machine tool servo system. The control system first decomposes the contour error into the normal tracking error and the advancing tangential error. A dynamic decoupled controller is then used to compensate the decomposed tangential and normal tracking errors. The normal controller minimizes the perpendicular tracking errors while the tangential controller maintains a desired feedrate.

Chen *et al.* [8] designed a contouring controller for a biaxial system in polar coordinates for which a relatively simple contour error model can be obtained. By using the techniques of feedback linearization from nonlinear control theory and robust control (such as sliding mode control), the resultant integrated controller combines the effects of feedback, feedforward and cross-coupled controllers.

Chiu and Tomizuka [13] showed that contouring performance can be viewed as a regulation problem in a moving task coordinate frame that is attached to a desired contour. By transforming the machine tool feed drive dynamics to this task coordinate frame, a control law is designed to assign different dynamics to the tangential and normal directions. The transformation also illustrated the effect of contour curvature and feedrate in the control action as well as the system dynamics in the task coordinate frame.

2.3.6 Synthesis of various control strategies

A new trend in the research of contouring control is the synthesis of various control strategies which were previously used separately. Feedback, feedforward and cross-coupled controllers all have their own advantages and disadvantages. By an integrated design, the contouring accuracy can be greatly improved. The aforementioned contouring controller proposed by Chen *et al.* [8] for a biaxial system in a polar coordinate system combines the effects of feedback, feedforward and cross-coupled controllers.

Yeh and Hsu [65] proposed an integrated controller for a biaxial motion control system which combines ZPETC and CCC to achieve both tracking accuracy and contouring accuracy. Studies also showed that ZPETC and CCC can be designed separately. An optimal ZPETC and a robust CCC were designed based on the contour error transfer function (CETF) to achieve desirable frequency responses and stability. In order to extend the available biaxial CCC to multi-axis motion systems, Yeh and Hsu [67] devised an algorithm to efficiently determine the variable gains for CCC by estimating the contour error vector for arbitrary curves. Later, Yeh and Hsu [68] proposed the extension of the integrated controller design to multi-axis motion systems as shown in

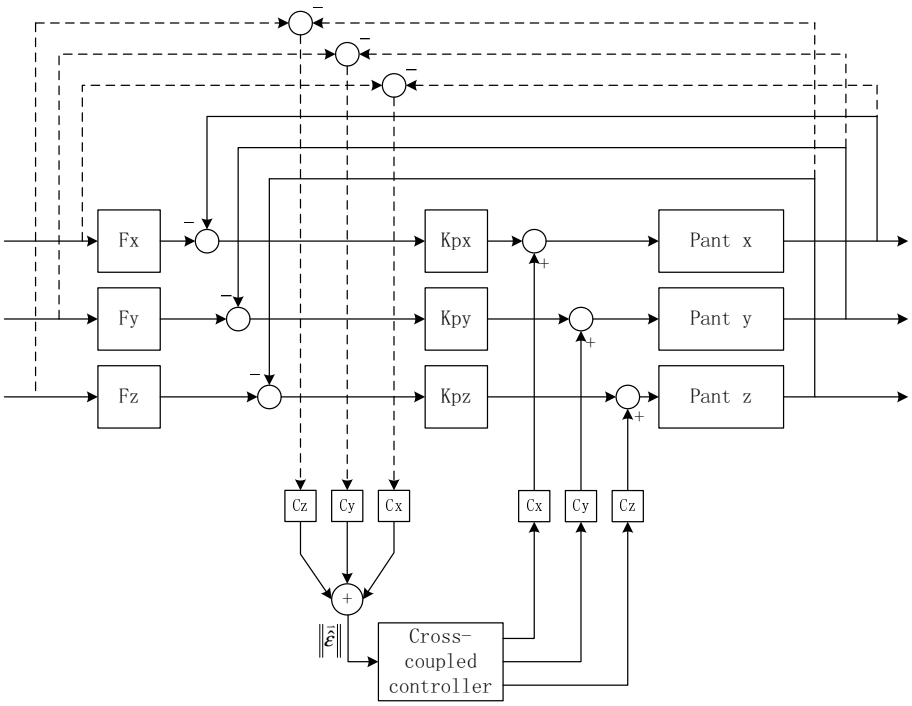

Figure 2.7: Multi-axis integrated control system [68].

Fig. 2.7, where Fx, Fy and Fz are feedforward controllers, and Kpx, Kpy and Kpz are feedback controllers.

2.4 Compensation

An alternative approach to improve the contouring accuracy is through compensation, which is to modify the reference position inputs to the axial control loops. According to their different nature, the compensation method can be classified into four types as shown in Table 2.1: (1) path precompensation; (2) iterative learning; (3) dynamic interpolation and (4) compensation for nonlinearities.

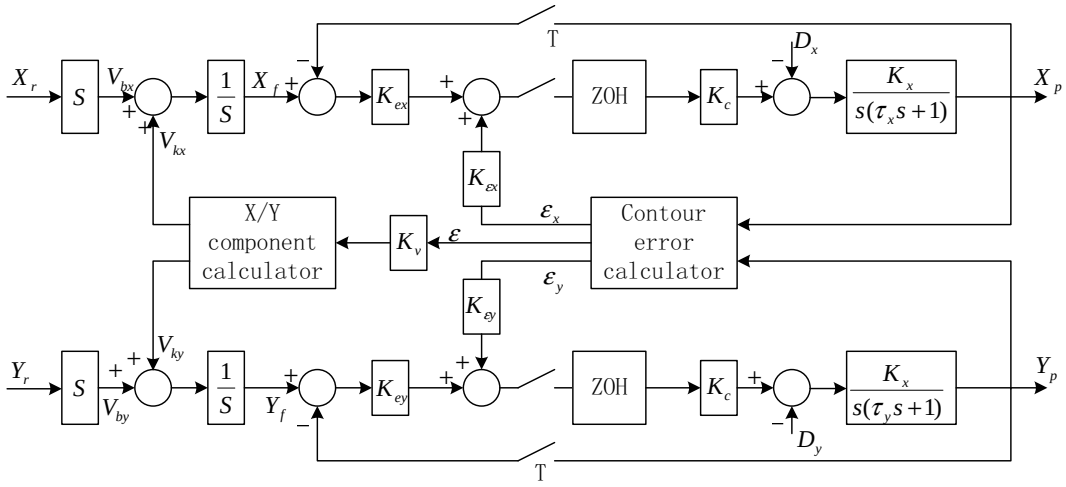

Figure 2.8: Block diagram of cross-coupled precompensation system [11].

2.4.1 Path precompensation

Chin and Tsai [12] proposed a path precompensation method (PM), and later introduced a cross-coupled precompensation method (CCPM) which combines the cross-coupling technique and the path precompensation method [11]. Chin *et al.* [10] used fuzzy logic to enhance the performance of CCPM. The structure of a system with CCPM is shown in Fig. 2.8. Note that if K_v, $K_{\varepsilon x}$ and $K_{\varepsilon y}$ are zero, then the system reduces to an uncoupled system. If the control gains $K_{\varepsilon x}$ and $K_{\varepsilon y}$ are set to zero with a nonzero K_v, the system reduces to a path precompensation method (PM) system. If K_v is zero with $K_{\varepsilon x}$ and $K_{\varepsilon y}$ nonzero, then the system becomes a typical cross-coupled control (CCC) system. Compared with CCC and PM, CCPM has been shown to be better in eliminating steady-state errors for linear contours and in reducing contour errors for circular contours. The advantage of CCPM is more evident at higher feedrates. At a feedrate of 200 mm/s the performance of CCPM stands out significantly [11].

The aforementioned path precompensation method utilizes *a posteriori* information. Corrections are made to the reference position inputs only after the contour error have occurred. If an error model can be used to predict the contour error when the reference position inputs are fed into the individual axial control loops, then modifications to the reference position inputs can be made based on the predicted contour error.

Ye *et al.* [64] proposed a cross-coupled path precompensation approach which first utilizes a prediction model to calculate tracking errors. Contour errors are then estimated based on these tracking errors. By a cross-coupling technique, the contour errors are used to calculate the compensation components along each individual axis. However, with the assumption that the tracking errors remain unchanged before and after compensation, this method is still unable to cancel out the contour error.

2.4.2 Iterative learning

CNC machine tools are commonly used to produce, not just a single part, but many parts in one set-up. For this repeating machining operation, contour error information collected in previous machining operations can be used for compensation so as to improve the performance of the current machining operation. This can be considered as offline compensation.

Lo and Hsiao proposed such a method of tool path compensation for repeated machining operations [34, 35]. The profile of the first machined part is measured by a coordinate measuring machine and, based on the measured data, the tool path is modified by a compensation algorithm and represented by a series of linear segments. The compensated tool path is then fed to the CNC machine tool for the machining

of subsequent parts.

The method proposed by Lo and Hsiao [34, 35] can be considered as a specific example of a broad class of techniques referred to as iterative learning control (ILC). Kim and Kim [25] investigated a PID type iterative learning controller for precise tracking control of industrial robots and of CNC machine tools performing repetitive tasks. The learning controller guarantees the convergence of the output error to zero under certain conditions even though the system parameters are not exactly known and in the presence of unknown external disturbances.

2.4.3 Dynamic interpolation

In [48] and [63], a dynamic interpolation method was proposed to improve contouring accuracy in real time. Dynamic interpolation is essentially based on an master-slave control architecture that uses a two-stage interpolator and two parallel communication channels [63]. The first communication channel is used to distribute coarse interpolation increments from the master and to communicate with the "slave" processors. The second communication channel (called state line) is used for synchronization, and to read and write the information about the state of all processors in the system. When the state line is high, then the interpolation is carried out as per normal. When the state line is low, then the interpolation just add zero increments to individual axial control loops. The low status of the state line can be triggered by process constraints. For example, in contouring systems, this constraint can be the axial tracking error. If the tracking error is beyond some specified value, this slave axial controller will change the state line to low. Because the logical operation for the state line is logical "AND", the state line will be held in the low state as long as any one of the slave

controllers sends a low signal. Through the use of this state line, the coordinated motion of different axes is achieved and the contour errors are reduced.

Later, this dynamic interpolation method was extended to a 3-dimensional case [40, 41]. Unfortunately, one of the disadvantages of dynamic interpolation is that an *ad hoc* hardware structure is required and this makes the technique difficult to be implemented on many existing CNC machines.

2.4.4 Effect of nonlinearities on contouring accuracy

In practice, there will invariably exist some kinds of nonlinearities, such as backlash, friction, saturation, and hysteresis in CNC systems. When the machine's axial controllers are designed based on linear models, these nonlinearities inevitably deteriorate contouring accuracy. Various methods have been investigated for overcoming the negative effects of nonlinearities on contouring accuracy.

Tarng and Cheng [53] found that the quadrant protrusions caused by stick-slip friction is a primary contour error in a circular test. It has also been demonstrated that appropriate tuning of the integral gain of the velocity loop can efficiently suppress these quadrant protrusions and torque disturbances on the circular profile.

Tarng *et al.* [55] developed a technique for the identification of and compensation for backlash to improve the contouring accuracy in CNC machining centers. Backlash error on the contouring profile is found as the direction of motion is reversed. A compensation strategy based on a simulated annealing optimization algorithm is then developed to reduce this backlash error.

Friction is one of the important factors resulting in contour error in the feed servo

systems of high speed, high precision CNC machine tools. Mei *et al.* [37] demonstrated that the friction error can be predicted offline. A friction error compensator can thus be designed and a control strategy for servo feed systems can be estimated with the precise mathematical model.

Hong [22] proposed an efficient identification technique for motion error sources using circular test results. The motion error signals from the circular test is classified into two fundamental patterns: nondirectional and directional error patterns. Frequency analysis and weighted residual analysis were used to identify the error sources of these nondirectional and directional pattern, respectively.

2.5 Problems in the literature

In the literature, however, there exist some problems remaining unsolved. The first is how to design a practically implementable algorithm to achieve axial matched dynamics. Yeh and Hsu [69] proposed a perfectly matched feedback control based on the identified model for multi-axis motion systems. Although perfect matched axial dynamics can be obtained theoretically, it is quite difficult to be achieved in practice because of inevitable modeling errors. To address this problem, a simple method of tuning gains to improve contouring accuracy has been proposed.

The second is the stiction compensation at quadrant positions. There are many methods dedicated to stiction compensation, which usually required detailed modeling of friction. The stiction compensation proposed in this thesis does not contain complex frictional dynamics modeling. Instead, it only requires a few intuitive parameter which can be easily obtained from an experimental test.

The third is to answer the question of how good contouring result a compensation scheme can achieve with the original controllers unchanged. The compensation methods reported in the literature can improve the contouring accuracy to some extent. However, there is no report that compensation can cancel out contour errors. Taylor series error compensation (TSEEC) basically states that with accurate axial dynamic models, the contour error can be totally canceled out with the compensation signals applied at the reference inputs of the servo loop.

The fourth question is how good tracking control be achieved, even under the conditions of model uncertainties, measurement noise and disturbances. With improved tracking control in each axis, the overall contouring accuracy can be improved for multi-axis applications. A new integral part design, a step-by-step design procedure and a choice of the sliding surface are all the improvements over previous studies.

Chapter 3

Factors affecting contour errors in CNC systems

This chapter investigates the effect of interpolation and encoder resolution, and mismatch in axes dynamics on contour errors in CNC systems. Simulation results show that axes dynamics mismatch are of importance to system performance in high accuracy systems. Contour errors are found to be highly sensitive to gain mismatch and the effect of this can easily outweigh all of the other effects.

3.1 Introduction

Because CNC machines are controlled by digital computers, their control is discrete in nature. According to the control architecture employed, CNC systems can be classified as either reference pulse systems or sampled-data (reference word) systems [28].

In the first type, the position control loop is external to the control computer and forms a digital *stepping* position control system as shown in Fig. 2.1. Inputs to

this digital stepping positioning system are normally in the form of either two pulse trains, one for each direction of motion, or a pulse train giving the number of steps of motion required and another input line giving direction information. In such systems, each input pulse represents one Basic Length Unit (BLU) of motion representing the *resolution* of the CNC system. The frequency of the input pulses gives the speed of motion along the respective axis and the number of input pulses the displacement. The maximum axial speed that can be achieved is limited by the maximum frequency of the pulses that can be generated by the interpolation routine.

In sampled-data CNC systems, at a pre-determined and constant *sampling frequency*, the position feedback for each axis is first sampled or read. This is then subtracted from the reference position command generated by the interpolator, normally in the form of a digital word, to obtain the axis position error. This error is then used to generate the control signal for the actuator or axis drive system. For sampled-data CNC systems, the reference position commands need only be generated at the same frequency as the sampling frequency by the interpolator. The output of this interpolator are digital words representing the desired reference positions of the axes. For sampled-data CNC systems, the maximum axial speed is not limited by the interpolation frequency as the position increment between two sampling instances could be any integral multiple of the BLU.

Although increasing the sampling frequency generally improves the position tracking errors, and thus the contouring accuracy, the sampling period, or the time available for any computations that is needed between sampling instances, is correspondingly reduced. This can become a problem if the interpolation, control and other algorithms

are computationally intensive, as in the case where complex error compensation techniques are to be employed.

The studies presented in this chapter shows clearly that the mismatch in the dynamics of the different positioning axes, and optimal axes gain settings have significant effect on contour errors when very high performance is required.

3.2 Contouring accuracy in reference pulse systems

The reference pulse system can be treated mathematically as a discrete-time system with nonuniform sampling periods [28]. The continuous motion of the axis is monitored by digital pulses from the feedback digital encoder, which produces a fixed number of pulses, K_e, per revolution. In practice, since K_e can commonly be larger than thousands of pulses per revolution, the system can be treated as a continuous-time system without much loss of accuracy. In this type of system, reference axes data in the form of pulses are required and the contour interpolation is commonly accomplished by a software Digital Differential Analyzer (DDA) [28]. The resolution of the DDA is determined by the length of its internal registers and will be the same as the feedback resolution of the digital encoder.

The reference axes position inputs are not continuous-time function but are staircase shaped, with the steps of the staircase equal to the resolution of the system. The reference pulse CNC system can thus be treated as a continuous control system receiving staircase reference inputs. In the simulation done here, the staircase reference inputs are generated by rounding off the position axes values generated to the

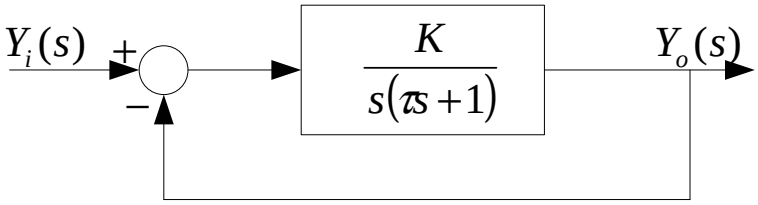

Figure 3.1: Model of individual axis.

required system resolution.

3.2.1 Matched and mismatched dynamics for reference pulse systems

In the work done here, a circular contour is used as the reference axes inputs for the study of contouring accuracies of the CNC system. As feedrate and the radius of the circle both affect contouring accuracies, the same feedrate of 11.8 mm/s with a radius is 30 mm (these parameters are the same as those used in [29]) are used for all the cases here. As mentioned above, the reference pulse system can be treated as a continuous-time system with staircase reference inputs. The model shown in Fig. 3.1 can be used to represent one axis of the CNC system. The value of the time constant τ used is 12 ms, which is in the typical range for CNC machines [28]. The closed-loop transfer function for this can be written as:

$$\frac{Y_o(s)}{Y_i(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n + \omega_n^2} \tag{3.2.1}$$

where

$$\omega_n = \sqrt{K/\tau} \tag{3.2.2}$$

$$\zeta = \frac{1}{2}\sqrt{\frac{1}{K\tau}} \tag{3.2.3}$$

Poo [43] has shown that the steady-state radial error for a constant feedrate, or angular velocity, along a circular contour and for a system with perfectly matched axes is given by:

$$\frac{e_r}{R} = \frac{1}{\sqrt{1 + (2\zeta\omega/\omega_n)^2 - 2(\omega/\omega_n)^2 + (\omega/\omega_n)^4}} - 1 \tag{3.2.4}$$

where ω and ω_n are the angular velocity of the travel around the circle and the undamped natural frequency, respectively. Their values are given by

$$\omega = \frac{F}{R} \tag{3.2.5}$$

$$\omega_n = \frac{1}{2\zeta\tau} \tag{3.2.6}$$

where F and R are the feedrate and the radius respectively. From Eq. (3.2.4), the damping ratio to achieve a zero steady-state radial contour error for a perfectly matched system can be obtained as

$$\zeta = \frac{1}{\sqrt{2(1 + \omega^2\tau^2)}} \tag{3.2.7}$$

For this, the value of the gain required is

$$K = \frac{1}{4\tau\zeta^2} = \frac{1 + \tau^2\omega^2}{2\tau} \tag{3.2.8}$$

Simulations were performed in the case of perfectly matched axes dynamics and with the gains set to achieve zero radial error. For the cases when the interpolation and encoder resolution are 1 and 10 μm, the resulting contour errors obtained are shown in Figs. 3.2 and 3.3, respectively.

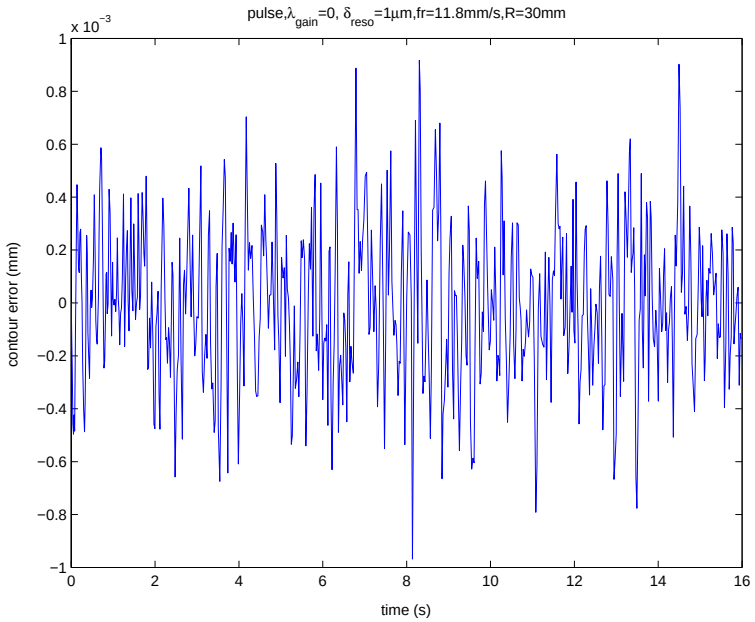

Figure 3.2: Contour error for matched dynamics. Resolution=1μm

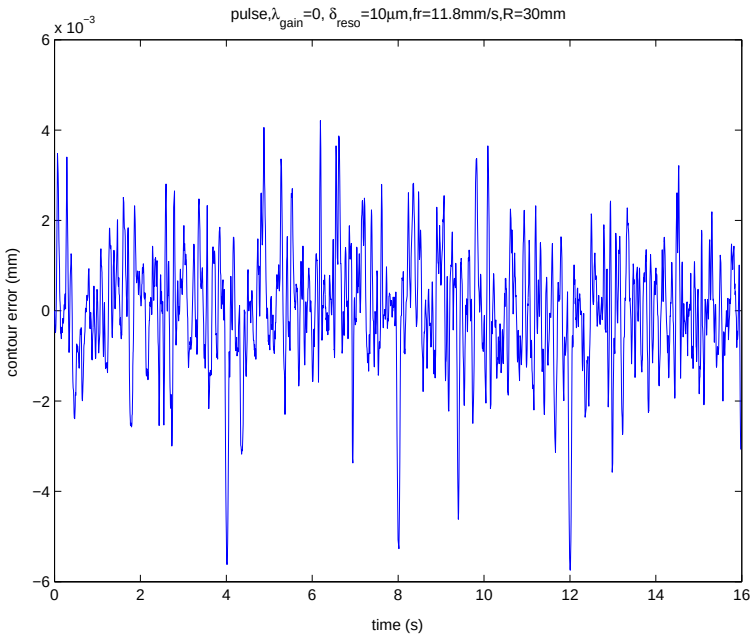

Figure 3.3: Contour error for matched dynamics. Resolution=10μm.

It can be seen that when the dynamics of the axes are matched and with the axes gains set at optimal values, increasing the interpolation resolution results in a corresponding increase in the contouring accuracy.

In practice, the dynamics of individual axes will be mismatched to a smaller or larger extent. Here, two types of axes mismatch are discussed - gain mismatch and time constant mismatch. The mismatch coefficients for gain and for time constant are defined as $\lambda_{gain} = \frac{\Delta K}{K}$ and $\lambda_{\tau} = \frac{\Delta \tau}{\tau}$ respectively. Simulation results for axial gain mismatch and time constant mismatch, with system resolution of 1 μm are shown in Figs. 3.4 and 3.5 respectively. It can be clearly seen that mismatches in axes gains have a far greater effect on contouring accuracy than mismatches in axes time constants. When there is a 1% mismatch in axes gains, the contour error is already greater than the 1 μm resolution of the system. For a 5% mismatch in gain, the maximum contour error is almost 8 μm. From Fig. 3.5, it can be seen that the contouring accuracy is far less sensitive to mismatches in time constants τ. A mismatch of 1% in τ has insignificant effect on the contouring accuracy. Even when there is a mismatch of 50%, the contour errors are still within the range of the 1μm resolution of the encoder. From these results, it can be concluded that gain mismatch, rather than time constant mismatch, is the major cause of contour errors.

3.3 Contouring accuracy in sampled-data systems

Unlike reference pulse systems, the position control loop in a sampled-data system is closed through the computer controller and control action is applied at a constant

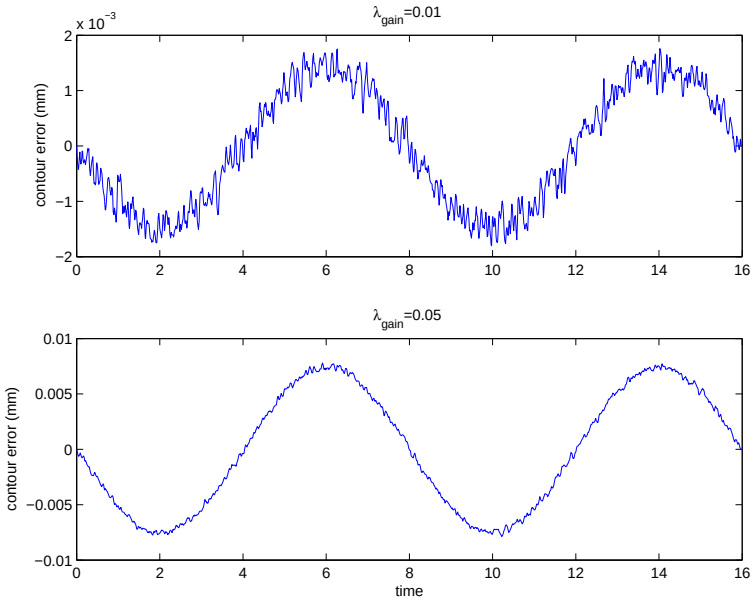

Figure 3.4: Contour error for mismatch in gain K.

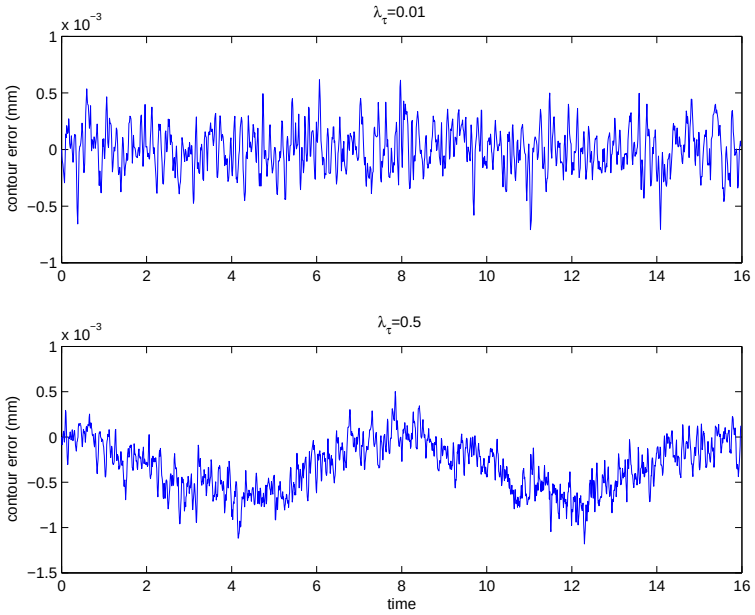

Figure 3.5: Contour error for mismatch in τ.

sampling frequency. For these systems, the reference position axes data inputs, generated at constant frequency by an interpolator, are in the form of digital words. The advantage of sampled-data systems over reference pulse systems is that the maximum axial speed that can be achieved is not limited by speed of the interpolator. This is especially important when very high resolutions, in the sub-micron range, are needed.

An important design parameter is the sampling frequency. Too low a sampling frequency may affect the system's dynamic accuracy performance while too high a frequency may result in there being insufficient time during a sampling period to perform all the required computations. In this study, a sampling frequency of 1 kHz is used.

3.3.1 Matched dynamics

First, the case in which the dynamics of the two axes are matched is considered. To better illustrate only the effects of optimal gain and acceleration interpolation, we first neglect the effect of the finite resolution of the interpolator and the digital encoder.

The contour error for a sampling frequency of 1kHz is shown in Fig. 3.6. It can be seen that, for matched dynamics, the contour error reaches a constant steady-state value after the transients.

Because of the effect of sampling and zero-order-hold, the dynamics of sampled-data systems is different from that of continuous-time system[5]. Eqs. (3.2.7) and (3.2.8) are thus no longer valid for determining the optimal gain for zero steady-state radial contour error. However, the steady-state contour error will still be related to the system gain and angular velocity. It is possible to tune the system gains, using a

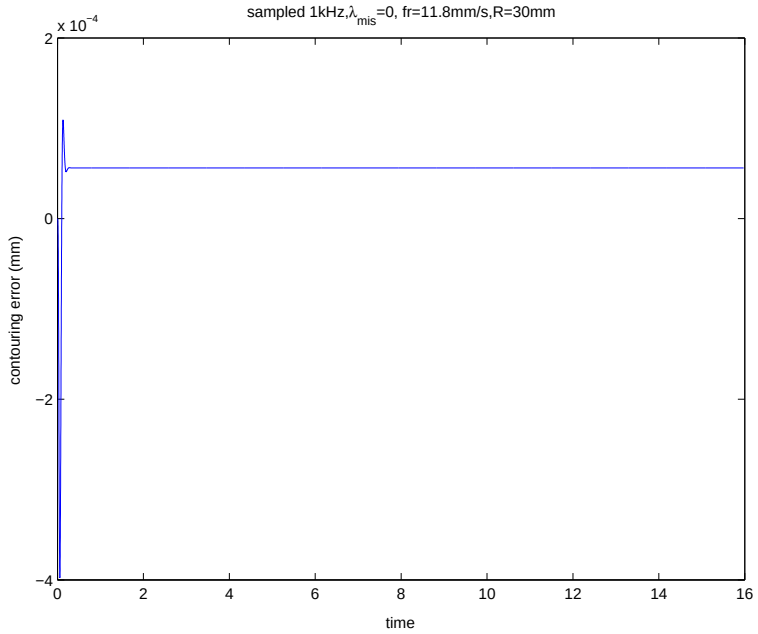

Figure 3.6: Contour error matched dynamics. Sampling frequency=1kHz.

heuristic search method, to find an optimal gain for minimum steady-state contour error. Using this method, an optimal gain of 40.00 was found which gives a steady-state error, for a sampling frequency of 1 kHz, in the order of 10^{-10} mm, as shown in Fig. 3.7. In part (b) of this figure, to better illustrate the smallness of the steady-state error, the initial transient portion of the error is not shown.

When this gain is kept constant and the sampling frequency is increased to 10 kHz, the resulting contour error is shown in Fig. 3.8. From this, it is clear that the optimal value of the system gain is a function of the sampling frequency as well as the angular frequency, which varies with the feedrate.

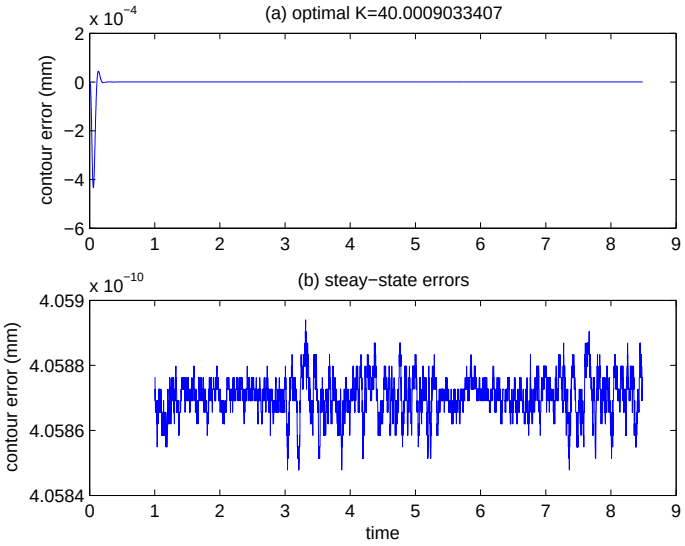

Figure 3.7: Contour error with optimal matched gains.

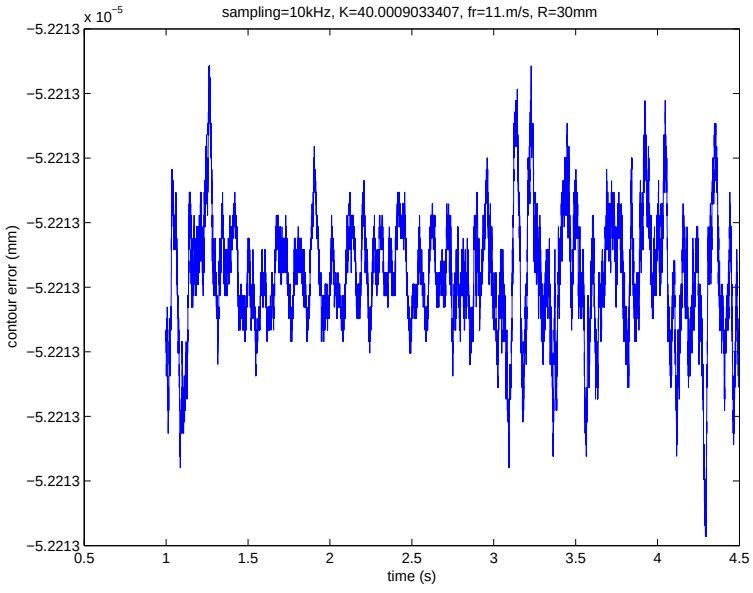

Figure 3.8: Contour error for non-optimal gains.

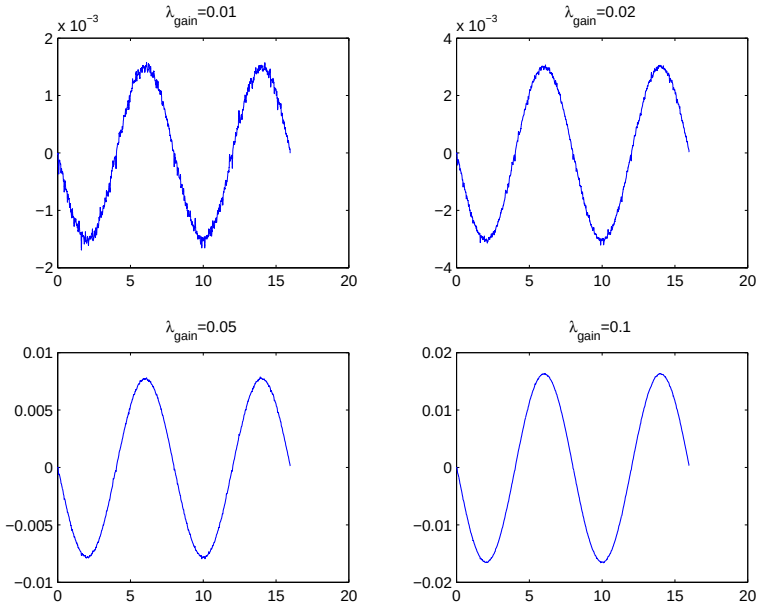

Figure 3.9: Effect of gain mismatch on contour error.

3.3.2 Mismatched dynamics

In practical situations, the dynamics of the different axes will not be perfectly matched and this will affect contouring accuracy. Two types of mismatch are discussed here - gain mismatch and time constant mismatch.

Four cases of gain mismatch, $\lambda_{gain} = 0.01$, 0.02, 0.05 and 0.1 were investigated and the results are shown in Figs. 3.9, 3.10 and 3.11. Similarly, four cases of time constant mismatch, with values of $\lambda_\tau = 0.1$, 0.5, 1 and 2, are studied and the results are shown in Figs. 3.12, 3.13 and 3.14.

Figs. 3.9 and 3.12 show the time response of the contour errors while Figs. 3.10 and 3.13 show the corresponding X-Y plots of these errors enlarged by 1000 times. From Figs. 3.11 and 3.14, it can be seen that the contour errors increase with increase in mismatch in both gains and time constants. The magnitudes of these errors indicate

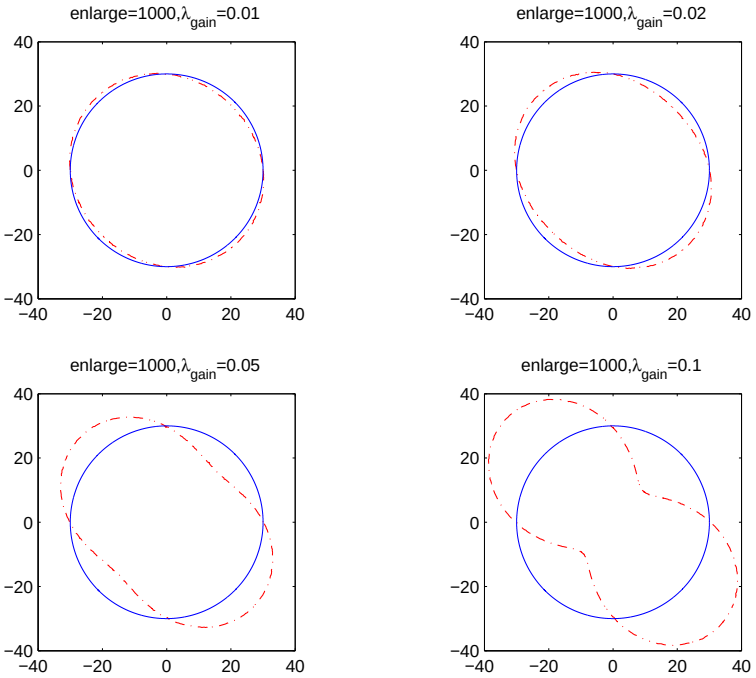

Figure 3.10: Effect of gain mismatch on contour error: X-Y plot with error enlarged by 1000.

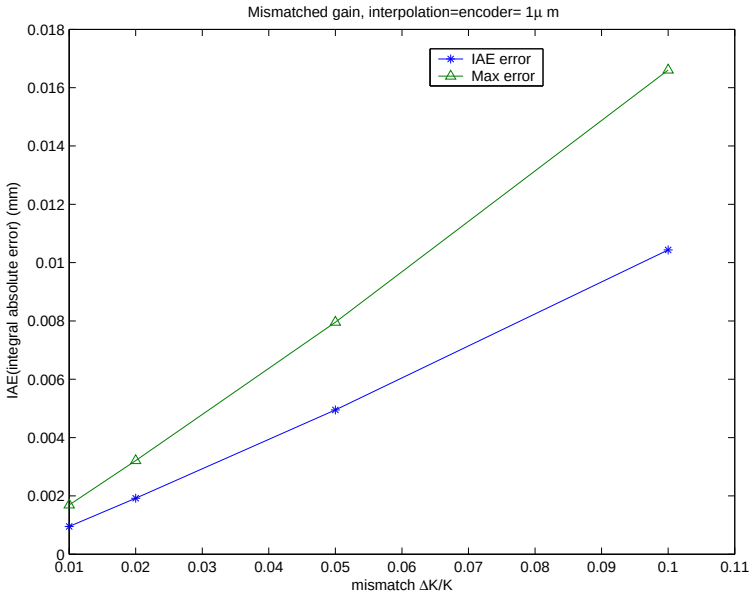

Figure 3.11: Effect of gain mismatch on contour error: IAE and maximum error.

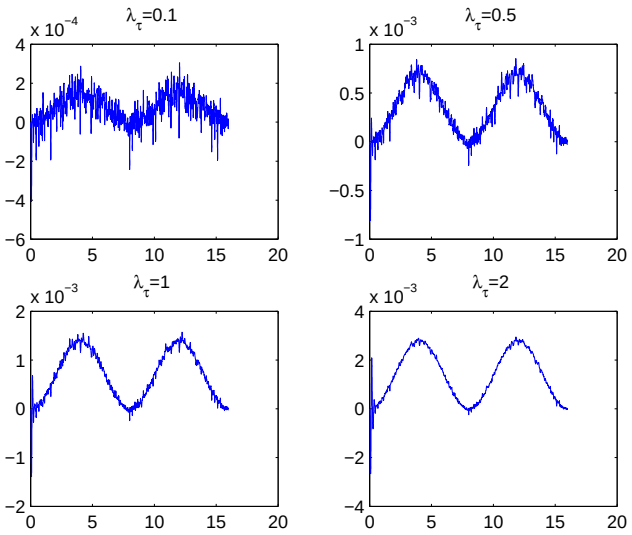

Figure 3.12: Effect of time constant mismatch on contour error.

that the effect of gain mismatch far outweighs the effects of other factors, such as sampling frequency, time constant mismatch, and system resolution when high accuracy is involved.

3.4 Conclusion

This chapter investigates the effect of system resolution and axes dynamics mismatch on contour errors in CNC systems. For systems with matched axes dynamics, it is found that system resolution affect contouring accuracy. Increasing the resolution of interpolation and the feedback encoders can improve contouring accuracy. However, this improvement is largely limited by the dynamics of individual axes. Simulation results show that contour errors are also affected by the mismatch of individual axes. For matched axes dynamics, an optimal gain can be found which can reduce the

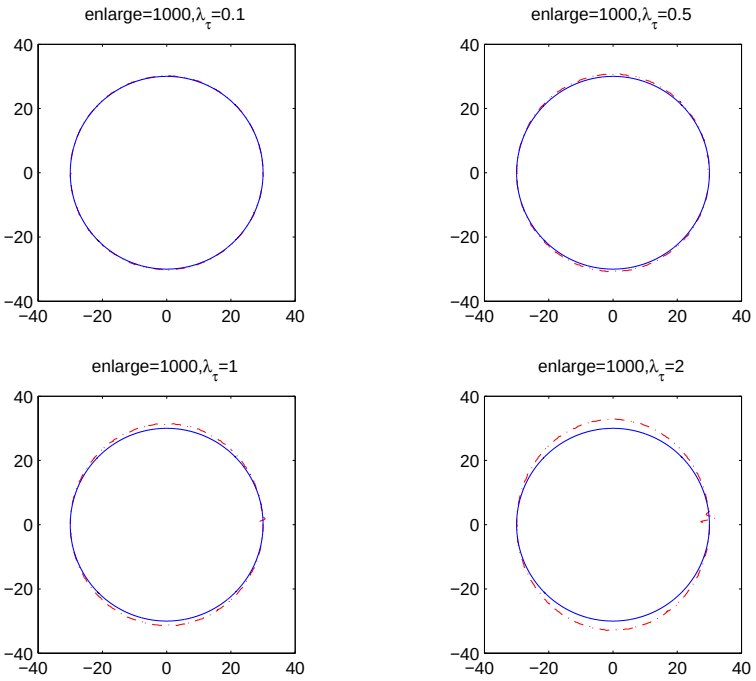

Figure 3.13: Effect of time constant mismatch on contour error: X-Y plot with error enlarged by 1000 times.

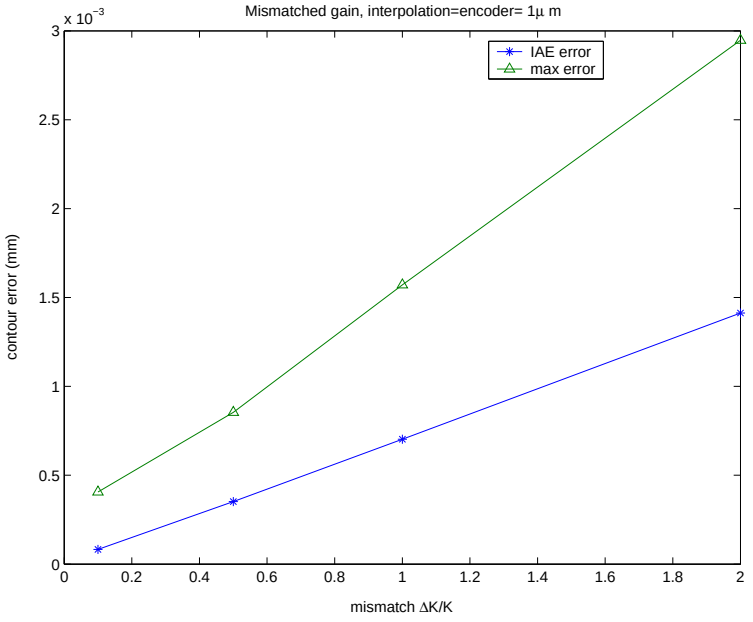

Figure 3.14: Effect of time constant mismatch on contour error: IAE and maximum error.

steady-state contour error for circular interpolation to nearly zero. In practice, there will be mismatch in axes dynamics and contour error is most sensitive to gain mismatch, which will be the likely dominating factor determining contour errors for high accuracy CNC systems.

Chapter 4

Improving contouring accuracy with matched axial dynamics

The contour errors usually result from unmatched axial dynamics and undesirable nonlinearities. The unmatched dynamics can be largely matched by tuning the proportional gains. Matched axial dynamics can, on one hand, effectively eliminate the linear contour errors, and, on the other hand, largely reduce the circular contour errors.

On the base of the matched axial dynamics, circular contouring accuracy can be analyzed from the frequency response of a linear system. Constrained by the bandwidth of the axial dynamics, there will be radial contour errors when a circular contour is generated. The radial contour error can be further reduced by feedforward compensation.

4.1 Introduction

Among the advanced controllers for feed drives which aim to improve contouring accuracy in CNC machines, there are the feedback, feedforward and cross-coupled controllers. In 1987, Tomizuka [56] proposed the zero phase error tracking controller (ZPETC) which reduces contour errors indirectly by reducing individual axial tracking errors. This was later improved upon [20, 58, 57]. The cross-coupled controller (CCC) originally proposed by Koren [27, 29] and subsequently improved upon [51, 49, 67], on the other hand, attempts to eliminate directly the contour errors. All the axes controllers, despite their various control strategies and structures, used the fundamental principle of using feedback to achieve good performance.

To improve contouring accuracy in general multi-axis motion systems, feedback controllers should be designed to achieve matched dynamic characteristics among all axes. Poo *et al.* [43] had shown the importance of dynamics matching in contouring applications. They showed that, for linear contours, when the axial dynamics are perfectly matched, the contour error will be zero. For circular contours, perfectly matched axial dynamics will result, in steady state, a perfectly circular contour with its radius either larger or smaller than the desired one depending on the angular velocity and system dynamics. Yeh and Hsu [69] proposed a perfectly matched feedback controller (PMFBC) design for multi-axis motion systems. Different axial dynamics were identified and the feedback controllers were then designed in order to perfectly matched the dynamics. However, because of inevitable errors in the identified models, perfectly matched dynamics cannot be achieved. Xi *et al.* [62] showed that the matching of axial loop gains is far more important than the matching of time

constants.

In this chapter, an approach is proposed which emphasizes the importance of keeping the axial dynamics simple by the use of a simple proportional controller. As compared with the more complex controllers which attempt to reduce tracking errors, a simple proportional controller will keep the axial dynamics simpler and, as a result, the matching of the axial dynamics can be more easily achieved. In this approach, the effort is focussed not on the reduction of tracking errors, but on the matching of these tracking errors in such a way as to reduce contour errors.

A simple approach is proposed to tune the proportional gains to achieve matched axial dynamics with as high a bandwidth as possible without exciting any undesirable vibration modes in the machine. Next, feedforward gains, applied at the reference position inputs, are further tuned to reduce radial contour errors for circular contours. The performance of this approach was evaluated experimentally on a two-axis mini-CNC machine with good results.

The rest of this chapter is organized as follows: Section 4.2 discusses the effects of axial dynamics on contour errors and the procedure for matching the axial dynamics by tuning the proportional gains and Section 4.3 introduces the experimental setup. The contouring performance with matched axial dynamics for both linear and circular contours are shown in Section 4.4. Next, Section 4.5 discusses the feedforward compensation for circular contours on the basis of matched axial dynamics and shows experimentally the effectiveness of feedforward compensation. Finally, Section 4.7 gives the conclusions.

4.2 Effects of axial dynamics on contour errors

4.2.1 Tracking errors for ramp inputs

The open-loop transfer function of a machine axis driven by a servo motor can be reasonably expressed as [29]

$$G(s) = \frac{K_m}{\tau s + 1} \tag{4.2.1}$$

In real machines, the axial dynamics are usually not matched as the inertial and frictional properties of different axes are different. These differences are reflected in the different time constants τ. For the model of Eq. (4.2.1), perfect matching of the axial dynamics will require perfect matching of both the proportional gain K_m and the time constant τ. Unfortunately, to match the time constants for the different axes is difficult if not impossible. As CNC machines are computer-controlled, changing the proportional gains is relatively easy.

In this chapter, the servo control loop of the mini-CNC machine used consists of an inner velocity control loop and an outer position control loop as shown in Fig. 4.1. The position controller used here is of the P (proportional) type. This is chosen for this study because of its simplicity based on the advantages stated before. The inner velocity loop, built into the servo motor drive that was used in this project, has a PI (proportional+integral) controller, which is commonly used. This model is thus used here for the theoretical analysis which follows rather than the simpler model used in Chapter 3.

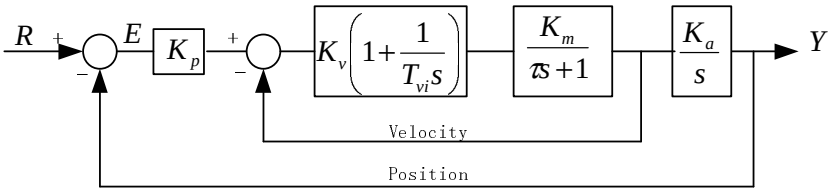

Figure 4.1: Block diagram for axial servo drive system.

K_x and K_y, the overall loop gains for the X and the Y axis respectively, are normally not known. However, for gain matching, their absolute values are not important. Matching the gains to minimize contour errors is a relatively straightforward process.

Except for the initial acceleration and deceleration, the velocity, or feedrate, along a contour in a CNC machine is normally constant.

Consider the model shown in Fig. 4.1. The error transfer function can be written as

$$\begin{aligned} G_e(s) &= \frac{E(s)}{R(s)} = \frac{1}{1+\frac{K_aK_pK_vK_m(T_{vi}s+1)}{T_{vi}\tau s^3+T_{vi}(1+K_vK_m)s^2+K_vK_ms}} \\ &= \frac{T_{vi}\tau s^3+T_{vi}(1+K_vK_m)s^2+K_vK_ms}{T_{vi}\tau s^3+T_{vi}(1+K_vK_m)s^2+K_vK_m(1+K_aK_pT_{vi})s+K_aK_pK_vK_m} \end{aligned} \tag{4.2.2}$$

For a ramp, or constant velocity, input $R(s) = V/s^2$, where V is a constant representing the desired velocity. We then have

$$\begin{aligned} E(s) &= G_e(s)\frac{V}{s^2} \\ &= \frac{V}{s}\cdot\frac{T_{vi}\tau s^2+T_{vi}(1+K_vK_m)s+K_vK_m}{T_{vi}\tau s^3+T_{vi}(1+K_vK_m)s^2+K_vK_m(1+K_aK_pT_{vi})s+K_aK_pK_vK_m} \end{aligned} \tag{4.2.3}$$

Using the final-value theorem [39], the steady-state tracking error is

$$e(\infty) = \lim_{s\to 0} sE(s) = \frac{VK_vK_m}{K_aK_pK_vK_m} = \frac{V}{K_aK_p} \tag{4.2.4}$$

As can be seen from Eq. (4.2.4), the motor gain K_m and velocity gain K_v do not affect the steady-state tracking error. This error is only dependent on the constant given by K_pK_a, with K_a being a lumped gain combining the effects of such parameters including the gains of the D/A (digital-to-analog) converters, amplifiers and other geometrical parameters. For the axes of a CNC machine, this gain can vary from machine to machine.

4.2.2 Errors for linear contours

To generate the desired linear contour at an angle of θ to the X axis and with a path velocity of V, the reference inputs to the two axes will be

$$R_x(s) = \frac{V\cos\theta}{s^2}, \qquad R_y(s) = \frac{V\sin\theta}{s^2} \tag{4.2.5}$$

where the subscripts x and y refer to the quantities associated with the X-axis and Y-axis respectively.

Referring to Fig. 4.2 in which P^* and P are the desired position and actual position respectively of the tool location along the linear contour during motion, the contour error at any given instance is given by

$$\varepsilon = -e_x\sin\theta + e_y\cos\theta \tag{4.2.6}$$

where e_x and e_y are the tracking errors along the X and the Y axes respectively. Using Eq. (4.2.4), the contour error in the steady state is given by

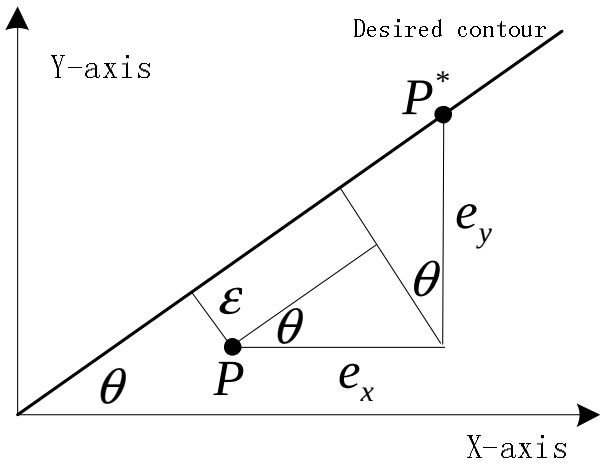

Figure 4.2: Linear contour error, P is the actual position, P^* is the desired position.

$$\begin{aligned} \varepsilon(\infty) &= -\frac{V\cos\theta}{K_{ax}K_{px}}\sin\theta + \frac{V\sin\theta}{K_{ay}K_{py}}\cos\theta \\ &= \frac{\sin 2\theta}{2}\left(\frac{V}{K_{ay}K_{py}} - \frac{V}{K_{ax}K_{px}}\right) \end{aligned} \tag{4.2.7}$$

It can be noted from Eq. (4.2.7) that if the loop gains of the two axes are matched such that

$$K_{ay}K_{py} = K_{ax}K_{px} \tag{4.2.8}$$

then there will be zero steady-state contour error. This is true irrespective of the velocity V along the contour or the angle θ of the line desired.

Furthermore, for a linear contour with $\theta = 45°$, the respective steady-state tracking errors will be

$$e_x(\infty) = \frac{V\cos 45^\circ}{K_{ax}K_{px}} \tag{4.2.9}$$

$$e_y(\infty) = \frac{V\sin 45^\circ}{K_{ay}K_{py}} \tag{4.2.10}$$

giving

$$\frac{e_x(\infty)}{e_y(\infty)} = \frac{K_{ay}K_{py}}{K_{ax}K_{px}} \tag{4.2.11}$$

Eq. (4.2.11) is used in the following section for the experimental matching of the axes loop gains.

In the above analysis, a slightly more complex third-order model was employed for the servomotor driven position axis. This was mainly because, in the drive for the motor that was used, a PI controller for the velocity loop was employed.

It can be easily shown that for the simpler second-order model shown in Fig. 3.1 in Chapter 3, identical equations (4.2.4) and (4.2.11) will also be obtained. In this case, only the value of the constants K_a, and thus K_{ax} and K_{ay}, will be different.

4.2.3 Procedure for matching loop gains

From Eq. (4.2.8), it can be seen clearly that if the loop gains of the two axes are equal, then zero contour errors in the steady state can be achieved. Eq. (4.2.11) shows that if the steady-state errors, $e_x(\infty)$ and $e_y(\infty)$, in following a linear contour at 45° can be experimentally measured, then the ratio of the existing axial loop gains can be

determined. Adjusting one, or the other, of the proportional gains within the control computer to match the loop gains can then be easily accomplished.

Assume first that the control gains of the individual axes of the machine have already been tuned such that they give good transient response without exciting undesirable higher-order dynamics. Assume further that, in this condition, the gains of the proportional controllers, as shown in Fig. 4.1, for the X- and the Y-axis are K_{pxold} and K_{pyold} respectively. Tuning these proportional gains to have matching axial loop gains can be easily done by first moving the machine to generate a linear contour at 45° to the X-axis at a reasonable path velocity. The axial steady-state tracking errors, denoted as e_{xss} and e_{yss}, are then measured. Eq. (4.2.11) is then made use of to reduce one or the other of the proportional gains in order to make the axial loop gains equal. Thus

1. If $e_{xss} > e_{yss}$, then the gains should be changed to

$$K_{pxnew} = K_{pxold} \tag{4.2.12}$$

$$K_{pynew} = K_{pyold} \cdot \frac{e_{yss}}{e_{xss}} \tag{4.2.13}$$

2. If $e_{xss} \leq e_{yss}$, then the gains should be changed to

$$K_{pxnew} = K_{pxold} \cdot \frac{e_{xss}}{e_{yss}} \tag{4.2.14}$$

$$K_{pynew} = K_{pyold} \tag{4.2.15}$$

The axial loop gains, after adjustment, will be matched and control contour errors for linear contours should be effectively zero. In matching the loop gains, the reason for reducing the gains of one or the other of the two axes rather than increasing them

is to avoid exciting undesirable higher-order dynamics which will cause errors due to vibration effects.

4.2.4 Circular contour errors

Circular contour errors are dependent on the frequency response of the axial dynamics. When a pure sinusoidal input is applied to a stable linear system, the output of the system in the steady state will also be a pure sinusoid with the same frequency as that of the input [39]. The magnitude and phase angle of the output, however, may differ from those of the input depending upon the dynamics of the system and the frequency of the input.

In generating a circular contour, suppose the inputs to the X and the Y axes are

$$r_x = R\cos\omega t, \qquad r_y = R\sin\omega t \tag{4.2.16}$$

The steady-state outputs for the X and the Y axes can be written as

$$x = M_x R\cos(\omega t + \phi_x), \qquad y = M_y R\sin(\omega t + \phi_y) \tag{4.2.17}$$

where M_x and M_y are the magnitude ratios and ϕ_x and ϕ_y the phase angles of the transfer functions for the X and the Y axes respectively. The values of both the magnitude ratios and phase angles are dependent upon the input frequency ω.

Due to unmatched axial dynamics, $M_x \neq M_y$ and $\phi_x \neq \phi_y$ and the resulting steady-state contour will be an ellipse. To obtain a perfect circular contour, the axial dynamics must be matched with $M_x = M_y$ and $\phi_x = \phi_y$.

With matched axial dynamics and a perfect circular contour, the resulting contour error may still not be zero as the generated contour may have a radius either larger or smaller than that desired. This radius depends on the frequency response of the axes and is dependent on the angular velocity of the generated circular path.

A straightforward method to reduce or eliminate this radial error is to feed forward a compensation term, in the form of a feedforward gain K_f, at the inputs. Suitable values for this gain, which will depend upon the frequency of the input, can be used to adjust the radius of the input reference circle so as to achieve zero contour error.

Without an accurate knowledge of the axial dynamics, an explicit expression for the required feedforward gain K_f will not be possible. Experimental determination of the required values of K_f, however, is a relatively straightforward process. Required values of K_f at various input frequencies can then be stored in a look-up table and used accordingly to reduce circular contour errors.

4.3 Experimental setup

Experimental evaluation of the proposed approach was carried out on a 3-axis mini-CNC as shown in Fig. 4.3 and the schematic diagram of the machine is shown in Fig. 4.4. Only the X and the Y axes were used. The control structure for one axis for this system is shown in Fig. 4.5. The motors used are Panasonic MSMA-042A1E

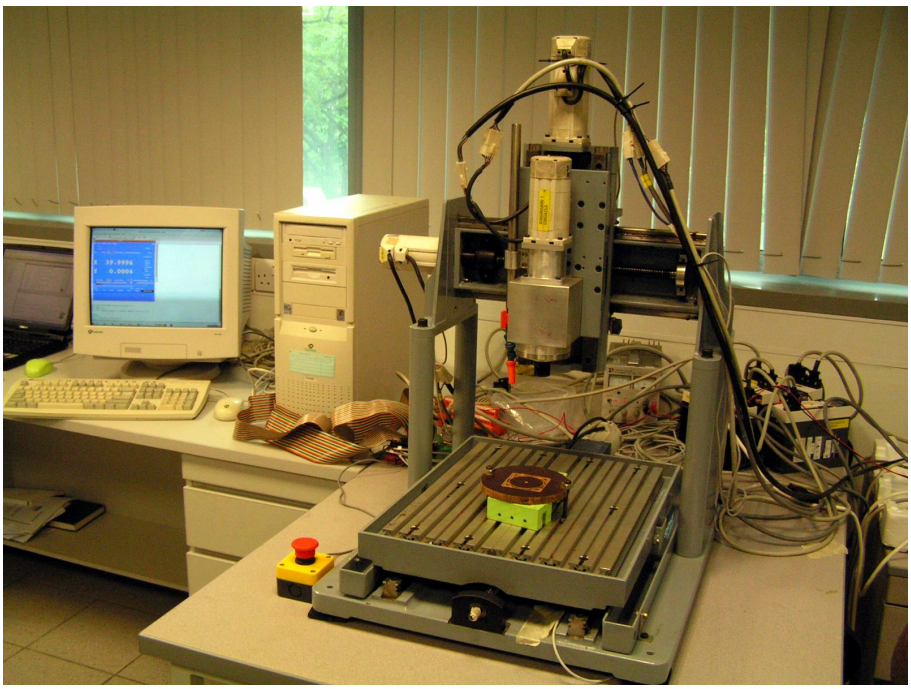

Figure 4.3: Picture of the 3-axis Mini-CNC.

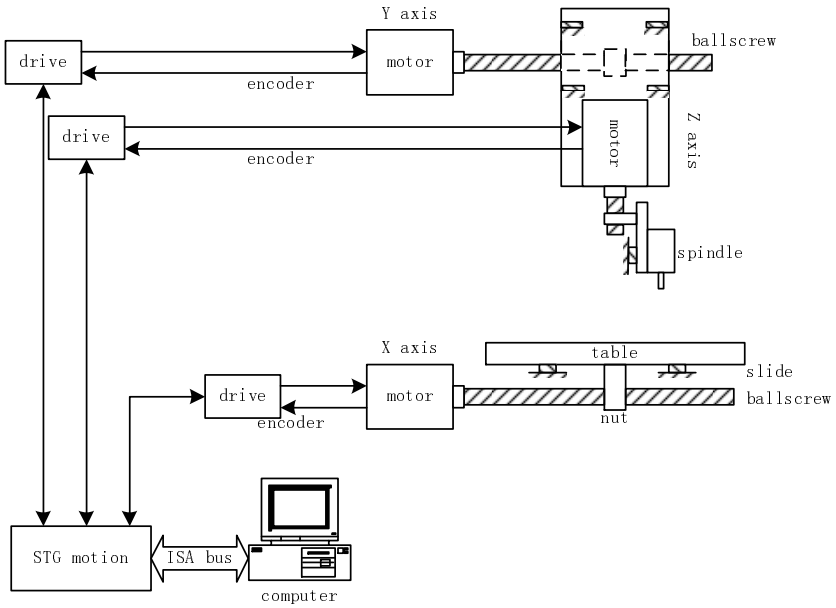

Figure 4.4: Schematic diagram for the 3-axis Mini-CNC.

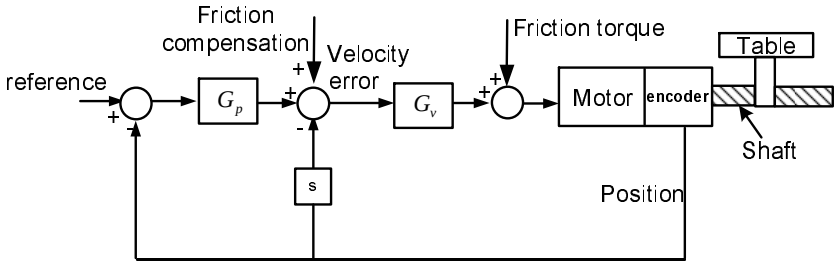

Figure 4.5: Schematic diagram for axial servo drive system.

AC servo motors with the motor drives set in velocity control mode. In Fig. 4.5, G_p is a proportional controller for the position loop implemented in the computer and G_v is a proportional-plus-integral controller for the inner velocity loop implemented in the hardware motor drive which accompanies the motor. Position feedback was implemented through an encoder which has a resolution of 10000 pulses/rev. The lead of the ball screw used is 4 mm giving a resolution for the position feedback as 0.4 μm/pulse of linear travel.

A Pentium III 450 MHz computer was used with a Servo-To-Go interface card. The computer operated under the RTAI real-time operating system which was patched to Ubuntu 6.06 Linux. The open source Enhanced Machine Controller (EMC) [1] was used as the CNC control software. The hardware abstract layer (HAL), a feature in EMC, was used to implement the real-time module used in EMC. The sampling frequency used is 1 kHz.

Table 4.1: Tuning of axial gains

K_{pxold}	K_{pyold}	e_x	e_y	$\gamma = e_x/e_y$	K_{pxnew}	K_{pynew}	e_{xnew}	e_{ynew}
11.0	10.0	0.1581	0.1736	0.9107	10.02	10.0	0.1736	0.1737
10.02	11.0	0.1736	0.1588	1.100	10.02	10.00	0.1736	0.1737

4.4 Performance with matched axial dynamics

4.4.1 Effect on linear contour errors

Following the tuning procedure proposed in Section 4.2.3, the proportional gains of the axes controllers were tuned in order to achieve matching loop gains. These gains, represented by K_{pxold} and K_{pyold} in the first row of Table 4.1, are 11.0 and 10.0 respectively.

When following a linear contour at 45° to the X-axis, the axial tracking errors in the steady state, e_x and e_y, are as shown in the first row of Table 4.1. Using these values and following Eqs. (4.2.12) to (4.2.15), the new values of axis gains, $K_{pxnew} = 10.02$ and $K_{pynew} = 10.00$ are computed.

With the new set of tuned gains, the system was made to follow the same linear contour at 45° to the X-axis. Fig. 4.6 shows the contour errors produced before tuning and after tuning. It can be seen that with the axial loop gains matched through tuning, the average contour error has been reduced to almost zero.

Fig. 4.7 shows the tracking errors of the individual axes after the gains have been tuned. It can be noted that even though these individual axial tracking errors are significant, because the axial dynamics have been matched, they cancel each other out and the resulting contour errors, as shown in Fig. 4.6, have effectively been reduced

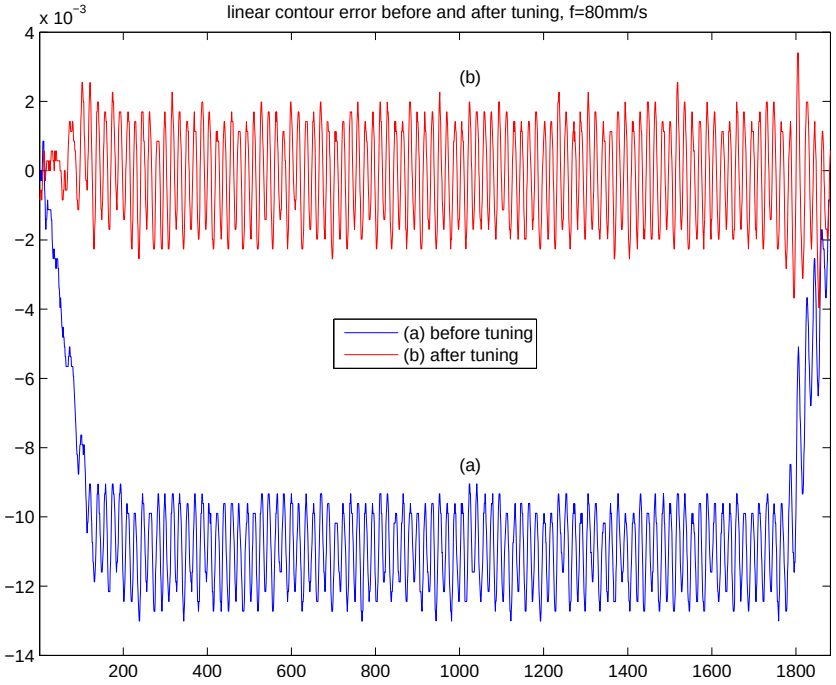

Figure 4.6: Linear contour errors before and after tuning.

Table 4.2: Linear contour errors for different angles after tuning the proportional gains (see Fig. 4.8)

θ (°)	ε_{max} (mm)	ε_{min} (mm)	ε_{mean} (mm)
30	0.001773	-0.002130	-0.000068
45	0.001696	-0.001699	0.000107
60	0.002336	-0.001862	0.000343

to zero.

As a confirmation check on the effectiveness of the proposed tuning procedure, the controller gain for the Y-axis was changed to 11 and the same tuning procedure repeated. The results obtained are shown in the second row of Table 4.1. It can be seen from the results that the axial dynamics have again been matched and the resulting tuned axial gains obtained are exactly the same as those obtained for the earlier case.

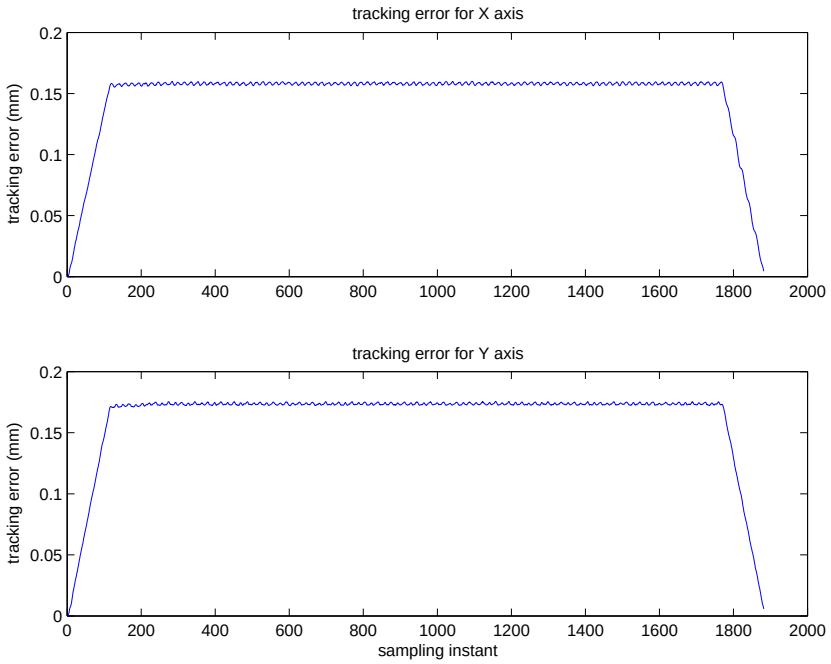

Figure 4.7: Tracking error for the X and the Y axes after tuning.

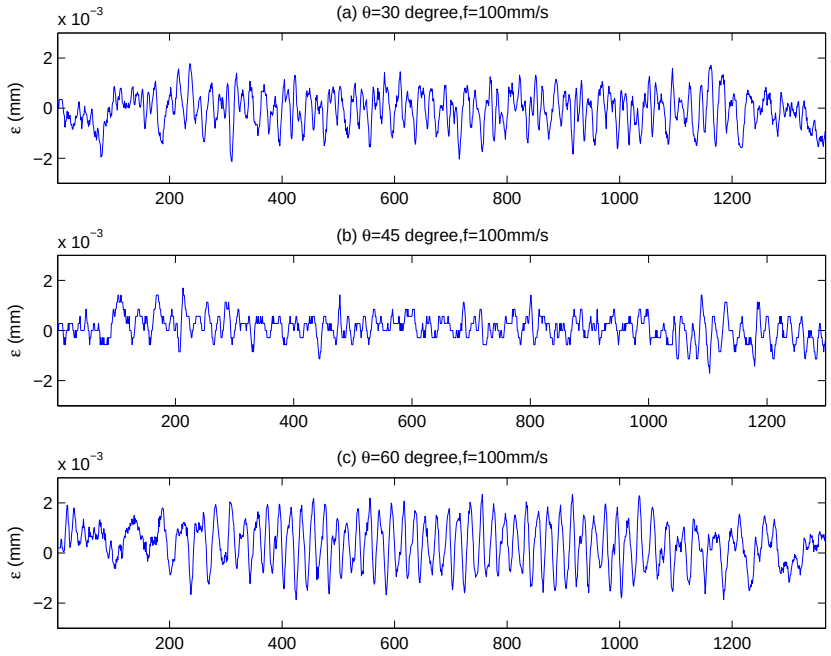

Figure 4.8: Linear contour errors for 3 different angle with respect to the X axis after tuning K_p: (a)30°, (b)45° and (c) 60°

After tuning the proportional gains, three linear contours were generated at three different angles with respect to the X axis, i. e., 30°, 45° and 60°. The results, shown in Fig. 4.8 and Table 4.2, indicate that the contour errors are within a band of ±2 μm, and the magnitude of the mean value of the contour error are all within the resolution of the encoder, which is 0.4 μm.

4.4.2 Effect on circular contour errors

As in the case for linear contours, the matching of loop gains is very important for circular contouring accuracy. Well-matched dynamics can significantly reduce the circular contour errors as compared with unmatched dynamics. Referring to Table 4.1, the three sets of gains are tested for the circular contouring accuracy for unmatched and matched dynamics. More specifically, the two unmatched cases (a) $K_{px} = 11$ and $K_{py} = 10$, and (b) $K_{px} = 10.02$ and $K_{py} = 11$ and the matched case (c) $K_{px} = 10.02$ and $K_{py} = 10$ are used to test the effect with a feedrate of 40 mm/s generating a circular contour of radius 40 mm, the contour errors are shown in Figs. 4.9 and 4.10.

It can be seen from these two figures that with unmatched gains (case (a) and (b)), the profiles of the contour errors are sinusoidal with a peak magnitude of about 6 μm. In the case of matched dynamics, case (c), this peak has been effectively reduced to zero. The remaining contour errors result from vibration and the effect of static friction at the quadrant positions. The profile of the contour errors in the X-Y plane for the aforementioned three cases are shown in Fig. 4.10, in which the magnitude of the contour errors has been magnified by 900 times. It is quite clear from the figure that the orientation of the contour errors in case (a) is just opposite to that of

case (b).

As a comparison, the contour errors obtained for matched gains (case (c)) were also plotted in Figs. 4.9 and 4.10. The contour errors in case (c) are also shown separately in Fig. 4.11 for better illustration. It can be seen that except at the quadrant positions where the contouring accuracy is adversely affected by the static friction, the average circular errors are very small. Except at quadrant positions, the profile of the contour error in case (c) is almost flat rather than sinusoidal as in the cases of (a) and (b). The flat average of the profile is still below zero, indicating that the actual circle is slightly smaller than the desired one.

The dominant contour errors appear at the four quadrant positions as there exist the reversal of velocities or the start from standstill at these points. The spikes at quadrant positions are as high as 6 μm for feedrate=40 mm/s. For case (c), we increase the feedrate from 40 to 100 mm/s and the contour errors are shown in Fig. 4.12. It can be seen that the profile of the contour errors is similar to Fig. 4.11, but the spikes are now as high as 7.5 μm. These spikes in contour errors are due to the effect of static friction.

Since the axial dynamics of the two axes are matched, the circular contouring accuracy is now mainly determined by the frequency response of the axes. More specifically, it is determined by the magnitude of the frequency response for the angular velocity if the static friction induced contour errors at the quadrant positions are not considered. In the range of achievable angular velocities, the magnitude of the frequency response should be unity as close as possible.

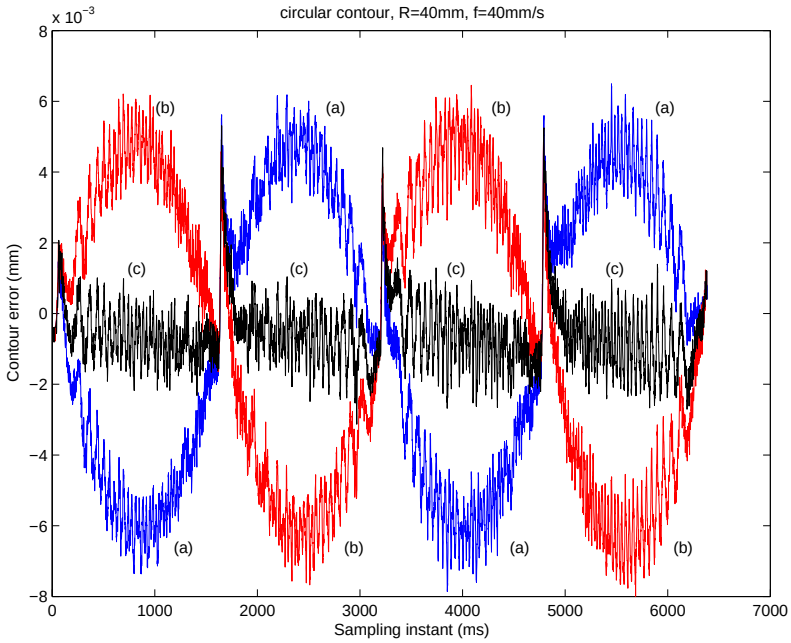

Figure 4.9: Circular contour errors for unmatched and matched dynamics: (a) $K_{px} = 11$ and $K_{py} = 10$; (b) $K_{px} = 10.02$ and $K_{py} = 11$; and (c) $K_{px} = 10.02$ and $K_{py} = 10$.

4.5 Compensating for radial error

After tuning the proportional gains K_p as discussed above, the dynamics of the X and the Y axes will be dynamically matched. As a result, the average linear contour errors for any straight-line path will be effectively zero and within the resolution of the encoder. Generated circular contours will also be circular. However, depending upon the velocity of motion along the circular contour path, there will be a non-zero average radial error.

The value of this average radial error depends on the frequency response characteristics of the axial dynamics, or specifically, the magnitude ratio of the frequency

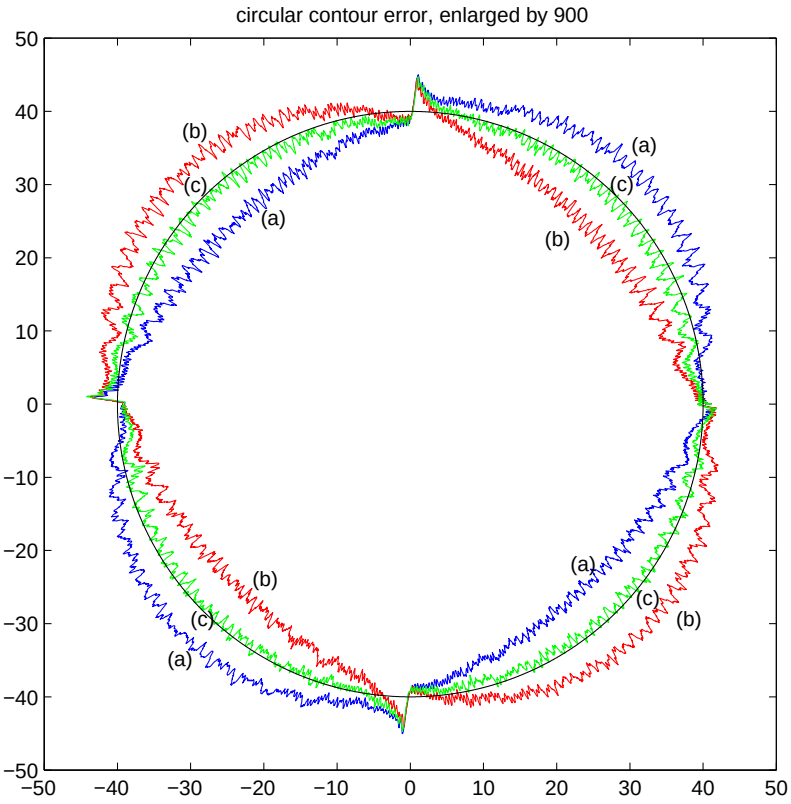

Figure 4.10: X-Y plot of circular contour errors enlarged by 900 for unmatched and matched dynamics:(a) $K_{px} = 11$ and $K_{py} = 10$; (b) $K_{px} = 10.02$ and $K_{py} = 11$; and (c) $K_{px} = 10.02$ and $K_{py} = 10$.

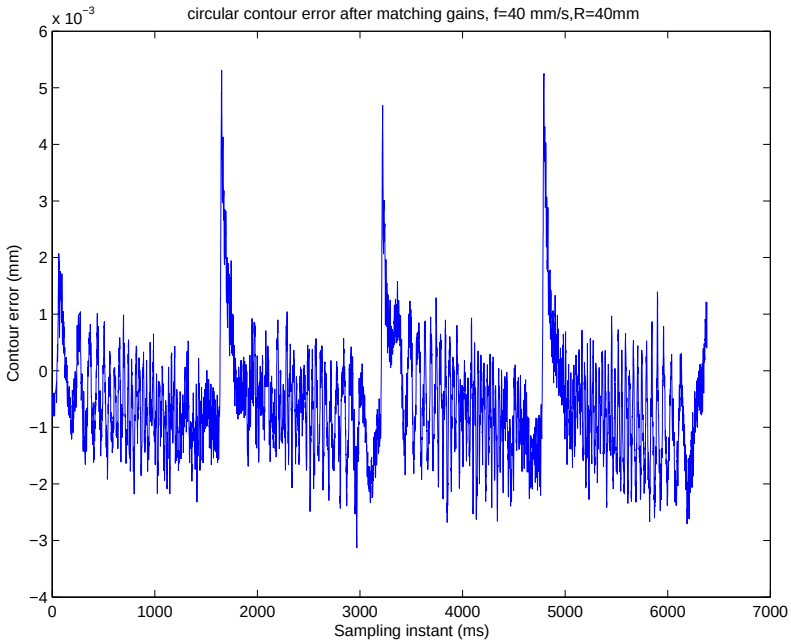

Figure 4.11: Circular contour errors for matched dynamics, $K_{px} = 10.02$ and $K_{py} = 10$, f=40mm/s,r=40mm.

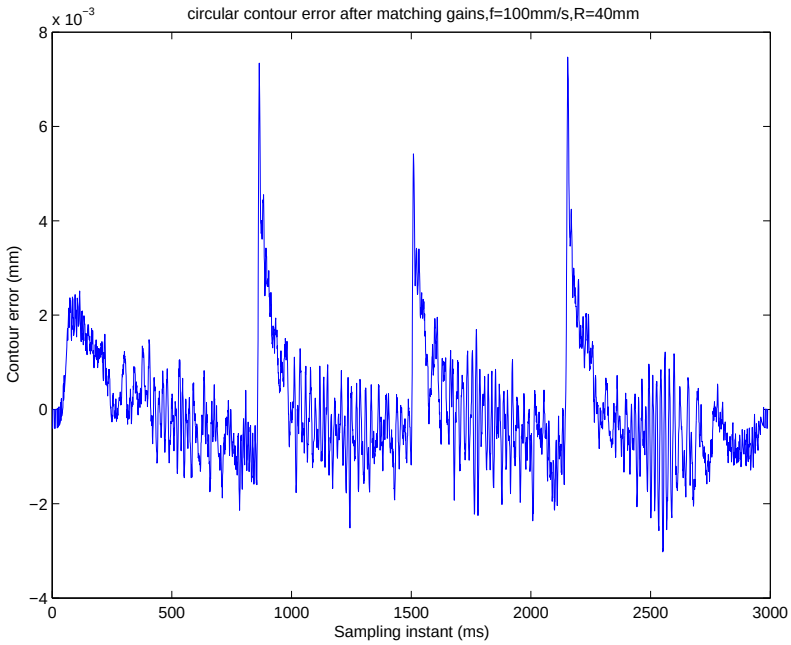

Figure 4.12: Circular contour errors for matched dynamics, $K_{px} = 10.02$ and $K_{py} = 10$, f=100mm/s,r=40mm.

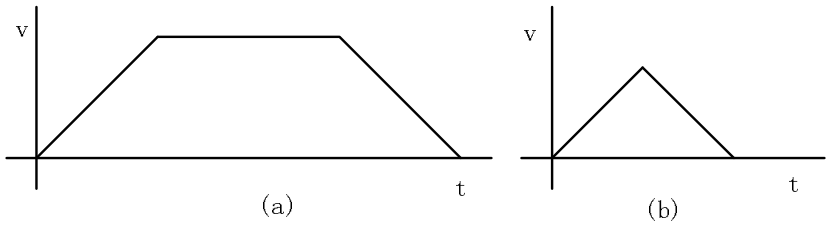

Figure 4.13: Interpolation with trapezoid velocity profile.

response at the angular velocity along the circular path. For any frequency, or angular velocity along the circular path being generated, the magnitude ratio representing the ratio of the radius of the generated circular path to that desired will be constant. This fact can be made use of to design a scheme to compensate for this error and will be discussed in the next section.

The other source of significant error is the static friction which occurs at the reversal of velocities or the move from standstill. This causes significant contour errors at the start of the circular path and at the quadrant positions where one axis reverses direction. A scheme for compensating for static friction will be discussed in detail in a separate chapter.

4.5.1 Achievable angular velocity

As the magnitude of the frequency response depends on the angular velocity, it is necessary to investigate the achievable angular velocities in a CNC machine. The physical constraints of a CNC machine limit the achievable maximum velocity and maximum acceleration/decelleration for each individual axis. Running a certain contour requires coordinated movements of involved axes. The combination of the constraints on individual axes gives rise to the constraint on the achievable maximum

velocity and maximum acceleration/deceleration for a certain trajectory. The achievable maximum velocity and acceleration along any path will need to be constrained to that for the weakest axis.

The maximum path velocities and acceleration are also subject to the maximum velocity and maximum acceleration which the interpolator can achieve. The interpolation in EMC adopts the simple trapezoid velocity profile for both linear and circular contours [1]. For circular interpolation, the velocity refers to the tangential velocity. For any path, the velocity profile consists of an acceleration, a cruise and a deceleration segment if the distance traveled is long enough to have a cruise segment as shown Fig. 4.13 (a). Otherwise, if the traveled distance is too short, the velocity is accelerated till half of the distance to be traveled before reaching the maximum velocity, then decelerated to zero from this point as shown in Fig. 4.13 (b). For linear contours, the maximum achievable velocity is determined by the maximum trajectory velocity. For circular interpolations, another constraint is that the achievable maximum angular velocity is not only determined by the maximum trajectory velocity, but also determined by the maximum acceleration and the radius of the circular path. For the circular interpolation in EMC [1], the achievable tangential maximum velocity is

$$v_{tm} = \min(v_m, \sqrt{a_m R}) \tag{4.5.1}$$

where v_m and a_m are the maximum trajectory velocity and acceleration respectively. Therefore, the maximum achievable angular velocity is

[1] as true for EMC 2.2.2

$$\omega_m = \frac{v_{tm}}{R} = \min(\frac{v_m}{R}, \sqrt{\frac{a_m}{R}}) \tag{4.5.2}$$

4.5.2 Feedforward compensation for radial contour error

As discussed in Section 4.2.3, the proportional gains for each axis, K_p, are first tuned so as to give good individual dynamic responses, typically with damping ratios between 0.7 to 1.0 depending upon how much overshoots can be tolerated. These gains are then adjusted, as described in Section 4.2.3, so as to achieve matched axial dynamics.

In general, the frequency response of the axes after tuning would be almost flat for frequencies within the bandwidth of the system. Still, almost 0 dB does not mean 0 dB and radial contour errors can still be a problem for high-precision machines which are required to move at reasonably high speeds.

To generate a circular path with a constant path velocity, the desired inputs to each of the two axes will be sinusoids, as given in Eq. (4.2.16) with R determining the radius of the desired contour and ω determining the path velocity which is given by ωR. From linear systems theory, we know that at steady state with the given inputs, the outputs will also be pure sinusoids at the same frequency and as given in Eq. (4.2.17). With matched axes dynamics, the magnitude ratios and phase angles will be the same or, referring to Eq. (4.2.17),

$$M_x = M_y \qquad \text{and} \qquad \phi_x = \phi_y \tag{4.5.3}$$

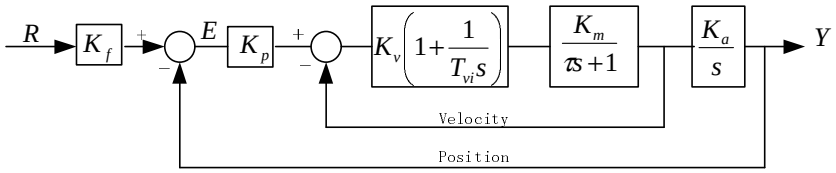

Figure 4.14: Feedforward gain compensation for circular contour.

so that perfectly circular contours will be generated.

The radial contour error will then depend only on the value of the magnitude ratio

$$M = M_x = M_y \tag{4.5.4}$$

If $M = 1$, there will be zero radial error. For linear systems, the value of M depends only upon the circular frequency ω and does not depend upon the magnitude of the input R. Based on this and with prior knowledge of the input circular frequency ω, a simple input compensation approach can be used to compensate for this radial contour error. This scheme is shown in Fig. 4.14, in which the value of the feedforward coefficient, K_f, which is dependant upon ω, is computed using a pre-calibrated look-up table. Values for this look-up table are easily obtained from calibration tests on the machine as will be elaborated later.

4.5.3 The feedforward compensation coefficient

The feedforward gain required for radial compensation, K_f, can be determined by actual experimental measurements on the target machine and the use of a look-up table.

In the experiment, suppose the machine is made to move along a circular path with a desired radius of R_d and at a desired path velocity of V_d. The ω_d, the desired angular velocity along the path, is given by

$$\omega_d = \frac{V_d}{R_d} \tag{4.5.5}$$

Suppose that the resulting circular path traced, in the steady state, has a measured average radius of R_a. Let the feedforward gain required for radial compensation, K_f, be written as

$$K_f = 1 + \beta(\omega) \tag{4.5.6}$$

where $\beta(\omega)$ is the required feedforward compensation coefficient. For linear systems, the magnitude ratio R_a/R_d will be a function only of the angular velocity ω, which can, for practical purposes, be taken as equal to ω_d. Consequently the required compensation coefficient, β, will also be a function only of ω.

Referring to Fig. 4.14, the compensated radius, R_c, at the output of the block K_f will be

$$R_c = K_f R_d \tag{4.5.7}$$

At the angular velocity of ω, the experimentally measured magnitude ratio is given by R_a/R_d. To compensate for radial error, we will thus need

$$R_c\left(\frac{R_a}{R_d}\right) = R_d \tag{4.5.8}$$

Using Eqs. (4.5.7) and (4.5.8), we have

$$(1+\beta)R_d\frac{R_a}{R_d} = R_d \tag{4.5.9}$$

giving

$$\beta = \frac{R_d}{R_a} - 1 = \frac{-\varepsilon}{R-\varepsilon} \tag{4.5.10}$$

As β is a function only of the angular velocity ω, experimental determination of β can be carried out at various values of ω_d and a radial compensation look-up table constructed. This loop-up table, together with interpolation between table entries if needed, can then be used to determine the compensation coefficient, and thereby the feedforward coefficient, K_f, needed to achieve radial compensation.

4.5.4 Experimental determination of feedforward compensation coefficient

The computation of R_a, the average radius of the circular path generated during experimental determination of the compensation coefficients, affects the accuracy of β. For a circular contour, the contour errors are affected not only by the axial dynamics, but also by the static friction at the reversal of velocities or the move from standstill. As the tuning of feedforward gains is meant to compensate for the frequency response,

in the steady state, of the axial dynamics based on linear systems theory, the contour errors caused by static friction should not be included in the calculation of R_a. For example, consider a circular contour starting from the point $(R, 0)$ w.r.t. the center of the circle $(0, 0)$ and moving in an anticlockwise direction back to the same point $(R, 0)$. To exclude the effects due to static friction, only the points on the circular path in the range of angles, $30^\circ \sim 90^\circ$, $120^\circ \sim 180^\circ$, $210^\circ \sim 270^\circ$ and $300^\circ \sim 330^\circ$ w.r.t. the X-axis are included in the computation of the average radius R_a. The segment of $330^\circ \sim 360^\circ$ is also excluded as it contains the deceleration stage. For the ranges included, the tangential velocity will be in the cruise stage and the contour errors will thus not be affected by static friction.

From linear systems theory, the feedforward compensation coefficient should be a function only of angular velocity. To verify that this statement is true, three sets of circular contours were run for three angular velocities (ω =1, 2 and 3 rad/s). For each set, three different radii (R=15, 20 and 25 mm) were used. The experimental results are shown in Table 4.3. From the table, it can be seen that for the same angular velocity ω, the values of β are about the same and vary with ω. The small differences in the values of β at the same ω, with a maximum difference of about 0.15×10^{-5}, would be due to disturbance effects such as vibrations.

The values of the compensation coefficient as a function of angular velocity can be expressed in a look-up table. A set of experiments are carried out to determine the feedforward compensation coefficient β for different angular velocities. As indicated in Eq. (4.5.2), with the same path velocity, a smaller radius can achieve a higher angular velocity. When R=15 mm, the contour errors obtained and the corresponding values of β measured at 5 different values of ω are shown in Table 4.4.

Table 4.3: same angular velocity and different radius

R (mm)	ω (rad/s)	v_{tm} (mm/s)	ε_{ave} (μm)	β (10^{-5})
15	1	15	-0.1115	0.7431
20	1	20	-0.1581	0.7907
25	1	25	-0.1612	0.6447
15	2	30	-0.1606	1.0706
20	2	40	-0.2324	1.1621
25	2	50	-0.2562	1.0247
15	3	45	-0.2511	1.6742
20	3	60	-0.3384	1.6922
25	3	75	-0.4516	1.8065

Table 4.4: same radius and different angular velocity

R (mm)	ω (rad/s)	v_{tm} mm/s	ε without K_f (μm)	β (10^{-5})	ε with K_f (μm)
15	1	15	-0.1115	0.7431	-0.0662
15	2	30	-0.1606	1.0706	-0.0713
15	3	45	-0.2511	1.6742	-0.0795
15	4	60	-0.3953	2.6355	-0.0075
15	5	75	-0.5012	3.3416	0.0082

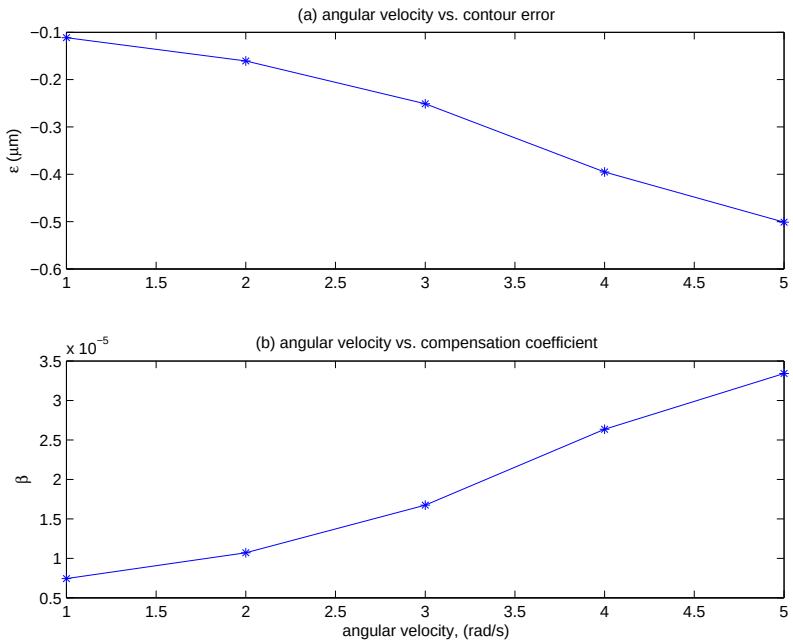

Figure 4.15: A circular contour R=15 mm: (a) angular velocity vs. contour error, (b) angular velocity vs. compensation coefficient

Fig. 4.15 (a) shows the plot of the radial error versus angular velocity. It is noticed that with the increase of angular velocity, the radial contour error becomes more negative, meaning that the resulting circle is smaller than the desired one. To achieve a zero contour error, it is necessary to increase the feedforward compensation coefficient as shown in Fig. 4.15 (b).

When a circle is generated, the steady-state angular velocity is reached in the cruise stage. The feedforward compensation is designed for the steady state. However, during the acceleration and deceleration stages, a smooth transition scheme need to be designed to avoid the discontinuity when β suddenly changes from zero to the value corresponding to the cruise stage. To avoid this, the value of β for any part of the circular contour in Eq. (4.5.6) is computed based on the estimated value angular velocity ω at that point rather than just a constant based on the cruise stage. For any value of ω which is not in the look-up table, a simple linear interpolation between the entries is used to compute the required value of β.

4.5.5 Performance with feedforward radial compensation

For the same five sets of conditions, with R=15 mm, used to experimentally determine the compensation coefficients β shown in Table 4.4, circular contours were generated with the feedforward compensation applied. The resulting average radial errors with compensation are shown in the 6^{th} column of Table 4.4. It can be noted that the magnitude of the average contour errors have been reduced to less than 0.1 μm.

With the calibrated table obtained at R=15 mm, experiments were conducted to determine the performance of the compensation approach at two other radii, at R=20

mm and R=40 mm. For these two radii, determination of the performance were also carried out at 5 angular velocities that were different from those used during calibration. These are done to test the performance when the value of β needs to be obtained by linear interpolation between entries in the look-up table. The results are shown in Tables 4.5 and 4.6. From the tables, it can be seen that, without compensation, the average radial errors increases with angular velocity ω and with the radius. In fact, one would expect, theoretically, that for any given angular velocity, the radial error would be proportional to the radius.

When radial compensation is applied, it is noted that the radial error are all reduced to with 0.1 μm, with the extent of reduction greater than one order of magnitude for larger values of ω when the errors without compensation are high. From Tables 4.5 and 4.6, it can be seen also that for intermediate values of ω, when interpolation was needed to compute the values of β, the radial error were similarly all reduced to within 0.1 μm, except for the case when v_{tm} = 140 mm/s. In this case, because of path velocity was very high, there were significant vibrations. This probably caused the larger error after compensation.

4.6 Contouring accuracy under machining

To test the contouring accuracy with matched axial dynamics under machining conditions, a linear contour from (0,0) to (10,10) mm, and a circular contour of radius of 5 mm were milling-machined on an aluminium part. The diameter of the tool cutter is 2 mm. The finishing cut was carried out with a feedrate of 5 mm/s, a RPM of 4500r/min and a cut depth of 2 mm. The contouring results are shown in Figs. 4.16

Table 4.5: average circular contour errors by feedforward gains, $R = 20$ mm

R (mm)	ω (rad/s)	v_{tm} mm/s	ε without K_f (μm)	ε with K_f (μm)
20	1	20	-0.1581	-0.0374
20	2	40	-0.2324	-0.0312
20	3	60	-0.3384	-0.0254
20	4	80	-0.5299	0.0088
20	5	100	-0.6710	0.0339
20	0.5	10	-0.1142	-0.0768
20	1.5	30	-0.1725	-0.0217
20	2.5	50	-0.2980	-0.0377
20	3.5	70	-0.4191	-0.0109
20	4.5	90	-0.6113	0.0277

Table 4.6: average circular contour errors by feedforward gains, $R = 40$ mm

R (mm)	ω (rad/s)	v_{tm} mm/s	ε without K_f (μm)	ε with K_f (μm)
40	1	40	-0.1773	0.0879
40	2	80	-0.3888	0.0077
40	3	120	-0.7526	-0.0890
40	0.5	20	-0.1300	-0.0047
40	1.5	60	-0.2752	0.0815
40	2.5	100	-0.5684	-0.0409
40	3.5	140	-0.7514	-0.1018

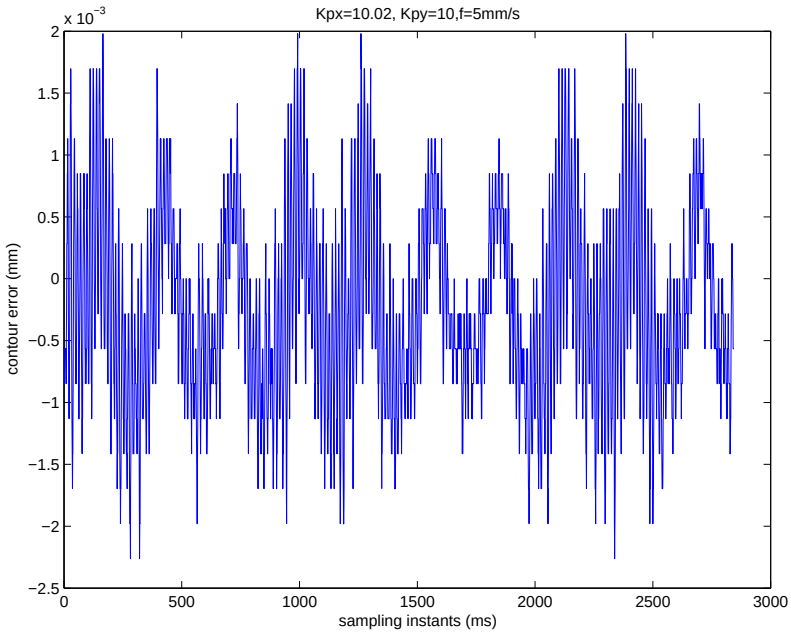

Figure 4.16: Machining a linear contour from (0,0) to (10,10)mm, feedrate=5mm/s, RPM=4500r/min and cut depth=2mm.

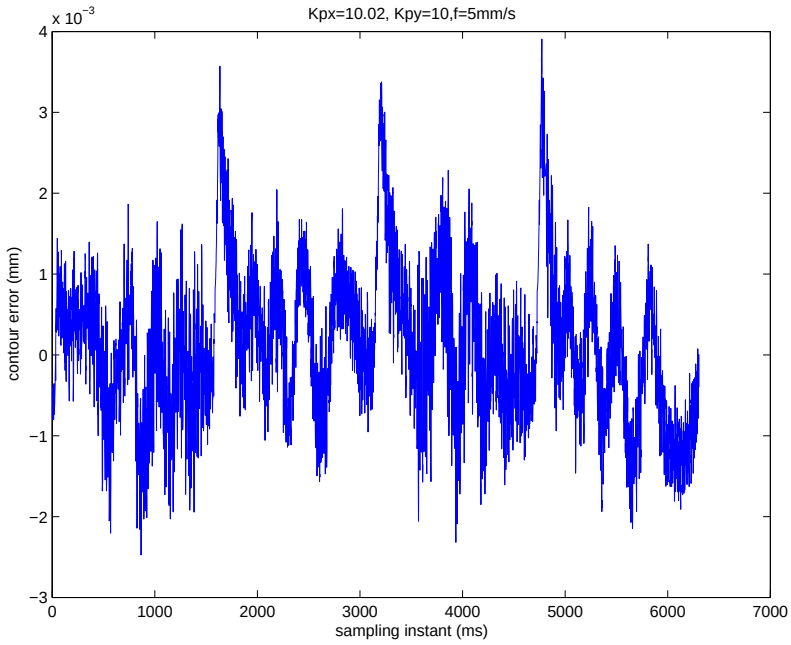

Figure 4.17: Machining a circular contour, R=5mm, feedrate=5mm/s, RPM=4500r/min and cut depth=2mm.

and 4.17 for the linear and circular contours respectively. Despite the vibrations and cutting forces in the machining, the contouring accuracy is still satisfactory. The linear contour errors shown in Fig. 4.16 within a band of about ± 2 μm, and the circular contour error are also within a band of about ± 2 μm, except that contour error spikes that are caused by the static friction.

4.7 Conclusion

Contour errors in continuous-path CNC systems usually result from unmatched axial dynamics and undesirable nonlinearities. The errors due to unmatched dynamics can be largely reduced by using a simple controller so as to keep the order of axial dynamics low and by tuning appropriately the proportional gains to match the dynamics. For any two axes of the system, the steady-state tracking errors of the two axes, measured when a 45 degree linear contour is generated, can be used to easily tune the proportional gains to match the axial dynamics. Experimental results show that with this simple approach, the resulting matched axial dynamics can, on one hand, effectively eliminate the linear contour errors, and, on the other hand, largely reduce the circular contour errors due to unmatched axial dynamics.

On the basis of the matched axial dynamics, circular contours will be made circular with accuracies greatly improved. However, because of limitation in the bandwidths of the axial dynamics, there will be radial contour errors and these can become significant with high path velocities, or, for any given angular velocities, with larger radii.

The radial contour error can be further reduced by a simple approach of feedforward

compensation. The feedforward compensation coefficient, which is a function of angular velocity, is determined experimentally for any given system and entered into a look-up table. Experiments conducted using this simple approach show the effectiveness of feedforward compensation in improving circular contouring accuracies.

With the aforementioned axial dynamics matching and feedforward compensation for radial errors, contour errors for both linear and circular contours can be greatly reduced. For the simple and not-so-well-built machine used in the experiments, average contour errors of lower than 0.1 μm (0.25 BLU) can be easily achieved. With these contour error improvement, it is noted that static friction induced contour errors now become the dominant errors. These errors occurred at the four quadrant positions in the circular path. In the next chapter, two methods for static friction compensation are proposed to overcome the static friction induced contour errors at the quadrant positions.

Chapter 5

Static friction compensation

In Chapter 4, we have seen the spikes in the circular contour errors at quadrant positions due to static friction effects at the reversal of velocities or the start from standstill. Quadrant positions refer to the positions along a circular contour whose angle with respect to the X-axis is a multiple of 90 degree, such as 0, 90, 180 and 270 degree. The stiction-induced spikes adversely affect the circular contouring accuracy.

To reduce the contour errors due to stiction, a compensation signal can be added to the control signal to the drive motor. In this chapter, a two-stage static friction compensation scheme is proposed, in which the compensating signal is a continuous signal comprising an increasing portion of a certain shape followed by a decreasing portion.

5.1 Introduction

In Chapter 4, a practical method of matching axial dynamics by appropriately tuning proportional gains on a real machine was proposed. Experimental results show that

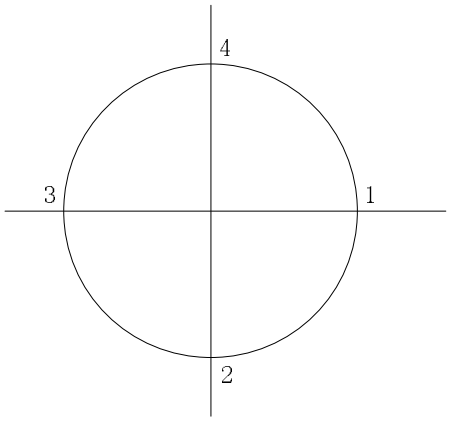

Figure 5.1: Quadrant positions of a circular contour.

the contouring accuracy for both linear and circular contours can be significantly improved. For circular contours, the radial errors resulting from the limited bandwidth of the axial dynamics can be further reduced by a simple method of feedforward compensation. As a result, the static friction (stiction) induced circular contour errors occurring at the four *quadrant positions*, points 1 to 4 in Fig. 5.1, become the dominant errors and these will need to be reduced if contouring accuracy is to be further improved. Static friction compensation schemes proposed here are aimed at suppressing the spikes in the circular contour errors at the four quadrant positions.

Friction itself is a very complicated phenomenon. Armstrong-Hélouvry *et al.* [4] made a comprehensive literature survey on friction characteristics and their analysis, and compensation schemes to reduce their effects which are of interest to the control community. A typical friction characteristic can be divided into four regimes, comprising static, boundary lubrication, partial fluid lubrication and full fluid lubrication friction. The commonly used model in engineering includes the static, Coulumb and viscous (or kinetic) friction as shown in Fig. 5.2. While it is commonly assumed

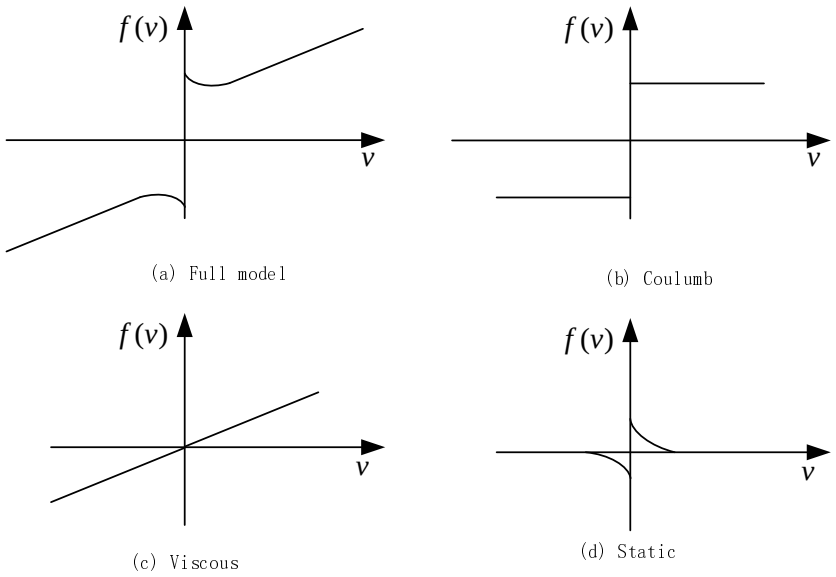

Figure 5.2: Full friction model and its components.

that there is no displacement during the static friction stage, i.e., before the applied force exceeds the break-away frictional force, research in tribology [3, 4] indicates that there are microscopic displacements during this presliding regime, in which the contact surfaces act as elastic bristles. The bristles will deform as long as the external force does not exceed the break-away frictional force.

As the external force increases, the deformation of the elastic bristles will increase much in the same manner as an elastic spring. When the external force exceeds the break-away force, the bonds between the bristles break. The breakaway point, at which the two contact surfaces begin to slide relative to each other, denotes the transition from the presliding regime to the sliding regime. At the breakaway point, the frictional force is at its maximum value. Thereafter, the boundary friction regime follows in which, as the velocity increases, the partial fluid lubrication friction takes over, and finally transits to the full fluid lubrication regime.

The transition from static friction to viscous friction is quite complex as illustrated by the negative-sloped viscous curve in part (a) and (d) of Fig. 5.2 . The negative-sloped viscous curve, usually referred to as the Stribeck curve, shows that as the velocity increases, the friction will decrease exponentially from static friction to viscous friction [3].

In CNC machines, it is quite common that the motor drives a table through a ball screw. As a typical mechanical transmission system, friction, especially static friction, degrades the circular contouring accuracy at the reversal of velocities and when moving from standstill. As a result, obvious spikes in contour errors are observed at the four quadrant positions. Static friction compensation can be used to reduce the effects of this static friction.

Various methods have been proposed to address the problem of static friction in motion control. Tarng and Cheng [53] demonstrated that by appropriately tuning the velocity loop integral gain, stiction-induced errors at quadrant positions can be effectively suppressed. Mei *et al.* [37] proposed a friction prediction model, and used this to tune a rectangular compensation pulse so as to overcome the stick-slip friction [38]. Park *et al.* [42] studied the characteristics of the presliding friction at velocity reversals. They estimated that the transition time from the presliding regime to the sliding regime is inversely proportional to the square root of the acceleration at zero velocity. The estimated transition time was then used as a criterion for applying different compensation schemes during different regimes. To avoid noise in the estimate of the acceleration, which was obtained as the second-order derivative of position with respect to time, they used the reference acceleration instead. Iwasaki *et al.* [23, 24] examined the dynamic characteristics of nonlinear friction using a friction

model with elastic bristles between the sliding surfaces of a table. The nonlinear friction model was found to behave as simple Coulumb friction under dynamic motion. An adaptive disturbance observer was implemented to realize the compensation for nonlinear friction. Tsai *et al.* [59] implemented a complete feedforward compensator to compensate for machine errors caused by servo lag and friction.

The servo control loop for the CNC machine used in this project consists of an inner velocity control loop and an outer position control loop as shown in Figs. 5.3 and 5.4. In the static friction compensation approaches employed here, a friction compensation component F_c is added to the inner velocity control loop as shown in Fig. 5.3.

In this chapter, a two-stage static friction compensation scheme with a continuous friction compensation signal is proposed and its performance is experimentally evaluated.

The rest of the chapter is organized as follows: Section 5.2 introduces a two-stage stiction compensation scheme. Section 5.3 and Section 5.4 discuss the determination of the breakaway displacement and the maximum compensation signal respectively. The displacement-based compensation is first discussed in Section 5.5, followed by the tracking error-based compensation scheme in Section 5.6. A conclusion is finally given in Section 5.7.

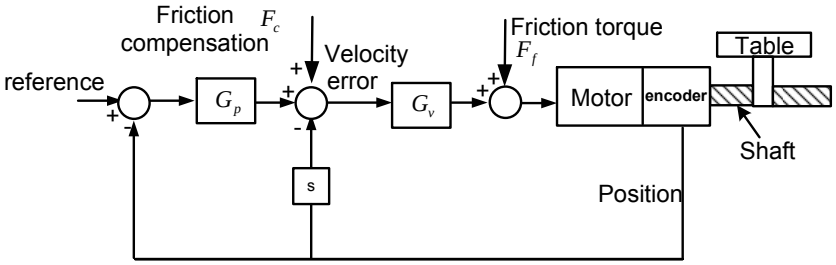

Figure 5.3: Schematic diagram for axial servo drive system.

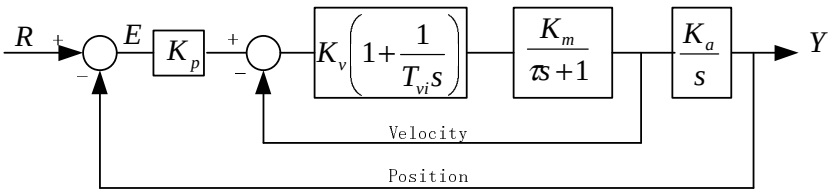

Figure 5.4: Block diagram for axial servo drive system.

5.2 Two-stage static friction compensation

5.2.1 Background

Referring to Fig. 5.2, there are three different types of friction, i.e. Coulumb, viscous and static friction. For simplicity, Coulumb friction, whose actual value is unavailable, is neglected in this project. Viscous friction, with its magnitude proportional to velocity, is already included in the model of a motor as in Eq. (5.2.1) [29].

$$G(s) = \frac{K_m}{s(\tau s + 1)} \tag{5.2.1}$$

As discussed in the earlier sections, during the presliding regime, the contact surfaces behave as elastic bristles and deform. This displacement between the surfaces before the breakaway point is referred to as the *presliding displacement*. Armstrong-Hélouvry

et al. [4] reported that the magnitude of the presliding displacement is in the range of $2-5$ μm. The maximum presliding displacement is called break-away displacement. When the break-away displacement is reached, the static friction is at its maximum value. After the break-away displacement, the static friction drops rapidly to zero, commonly referred to as the Stribeck effect. Armstrong-Hélouvry *et al.* [4, Chapter 6] summarized empirical models that fit this decaying friction as a function of velocity. Two commonly used models are

$$F_{stribeck} = (F_s - F_k)e^{-v/v_s} \tag{5.2.2}$$

$$F_{stribeck} = (F_s - F_k)\frac{1}{1+(v/v_s)^2} \tag{5.2.3}$$

where F_s is the static friction, F_k the viscous (or kinetic) friction, v the current velocity and v_s the characteristic velocity (an empirical parameter).

As significant contour errors appear at the quadrant positions, the proposed friction compensation scheme is focused on static friction. More specifically, the compensation scheme is effective only in the transition from the presliding regime to the sliding regime. The scheme is basically a two-stage scheme. In the presliding regime, the friction is considered as a function of tracking error, while in the sliding regime, the Stribeck effect (negative viscous friction) is considered as a function of velocity and tracking error.

To study the effect of stiction on contour errors, the target mini-CNC first had its axes dynamically matched as described in Chapter 4. Contour errors without stiction compensation were obtained when two circular contours of radius of 40 mm were generated at two different feedrates, 40 mm/s and 100 mm/s. The results are shown

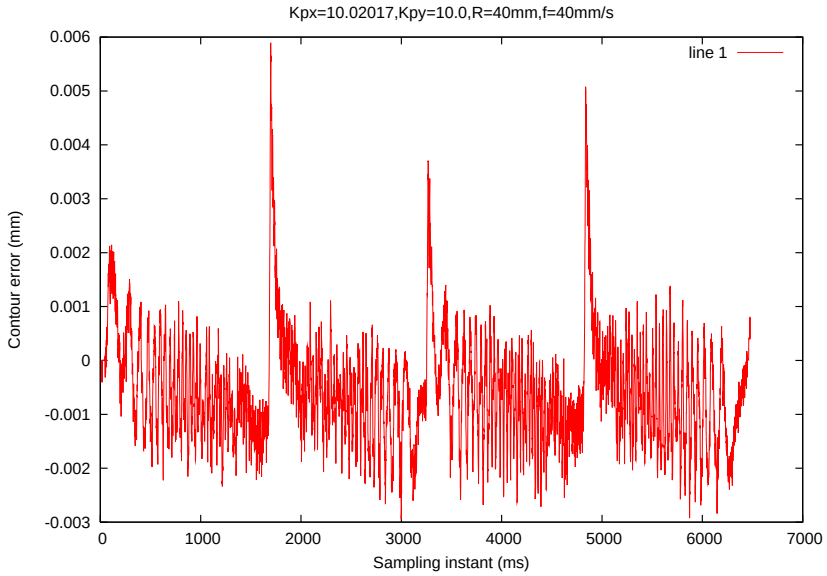

Figure 5.5: Contour error without friction compensation when f=40mm/s, R=40mm

in Figs. 5.5 and 5.6, respectively. It can be noted that the contour errors at quadrant positions ranged between 6 and 8 μm due mainly to the stiction effect. In comparison, it can be further noted that, apart from these quadrant positions, the contour errors at other parts of the circular contour are within 2 μm.

In the following schemes for stiction compensation, the compensation is applied after the machine has been properly tuned so that the axes are dynamically matched.

5.2.2 Two-stage continuous compensation

Compensation for static friction is divided into two phases. The first is during the presliding regime which starts from zero velocity to the moment when the breakaway displacement d_b is reached. This is followed immediately by the second phase for the Stribeck regime (the negative viscous friction regime) when the displacement from the

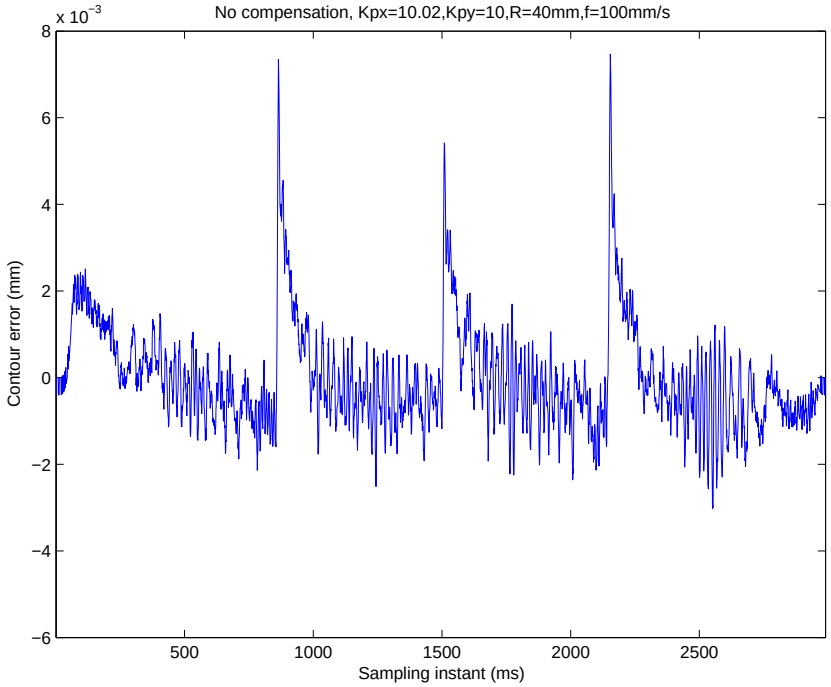

Figure 5.6: Contour error without friction compensation when f=100mm/s, R=40mm

point of zero velocity is larger than the breakaway displacement d_b. Compensation for static friction is initiated whenever zero velocity is detected in an axial motion. For better detection of this state, the reference velocity is used to determine the point of zero velocity as this is not contaminated by noise as in the measured velocity.

The compensation signal is designed as a product of a compensation coefficient, k_f, and a maximum compensation value, u_{max}, and given by

$$F_c = \text{sgn}(v_r) k_f u_{max} \tag{5.2.4}$$

where F_c is the compensation signal, v_r is the reference velocity, k_f a compensation coefficient with a value between 0 and 1, and u_{max} a constant parameter which is

experimentally determined for any given axis in any direction. The sign of F_c is the same as that of the reference velocity.

In the next few sections, we will discuss the choice of breakaway displacement d_b, the determination of u_{max} and the design of the compensation coefficient k_f both in displacement-based and in tracking error-based.

5.3 The breakaway displacement d_b

For the two-stage continuous compensation scheme, it is necessary to determine the breakaway displacement, d_b, defined as the maximum presliding displacement before sliding occurs. Armstrong-Hélouvry *et al.* [4] reported this to be in the range of $2-5$ μm. Wu and Tung [61] used a very slow ramp input (2 μm/s) to determine the breakaway displacement. They reported values of 4 μm for the forward direction and 5 μm for the backward direction.

As the values of d_b is limited to a small range of between $2-5$ μm and experimental determination is not very straightforward, in this project, a few values of d_b in the reported range were used in implementing the proposed compensation approach. Experiments showed that any value of d_b between the range of 3-5 μm works well.

5.4 Determination of u_{max}

Experimentally, the values of u_{max} are determined by moving both the X- and Y-axes from standstill along a 45° linear contour at a very low speed. In our experiments,

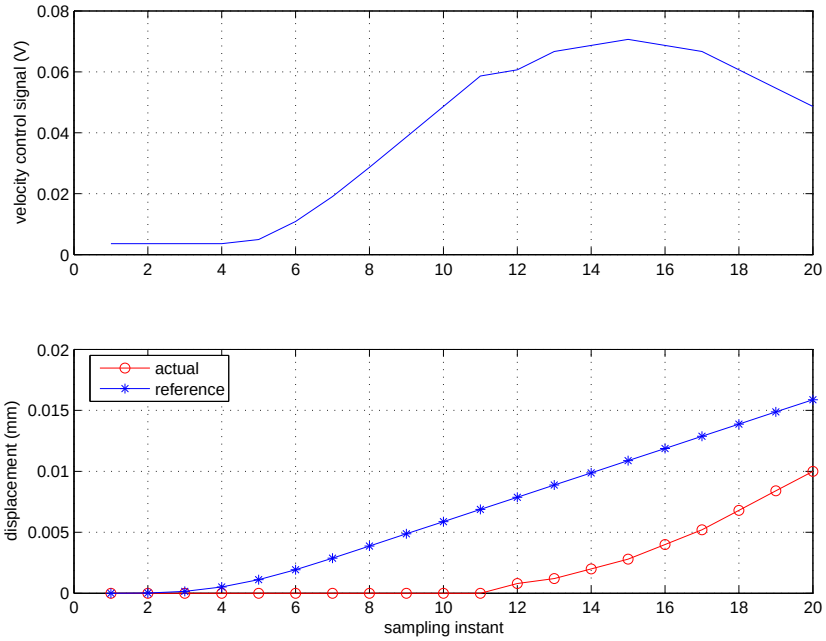

Figure 5.7: Positive u_{max} for X axis when f=1 mm/s.

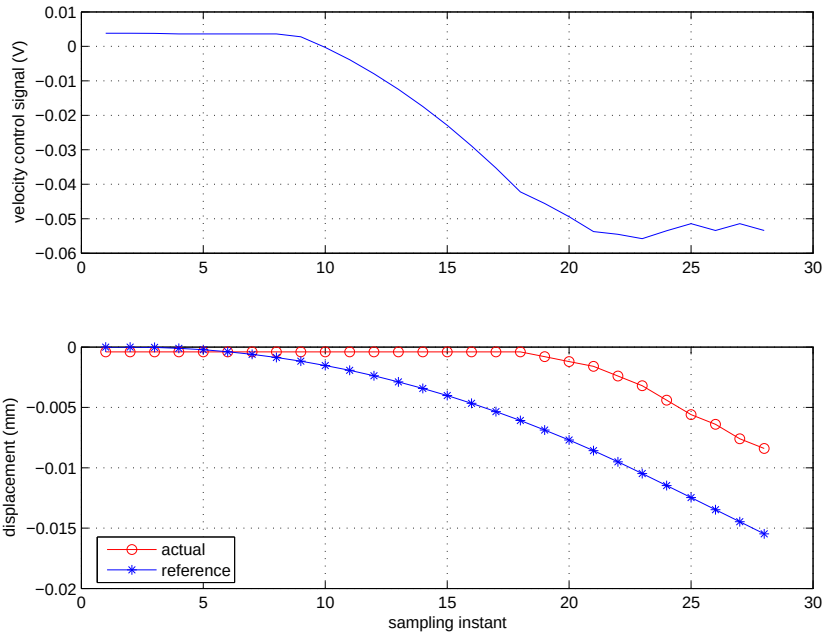

Figure 5.8: Negative u_{max} for X axis when f=1 mm/s.

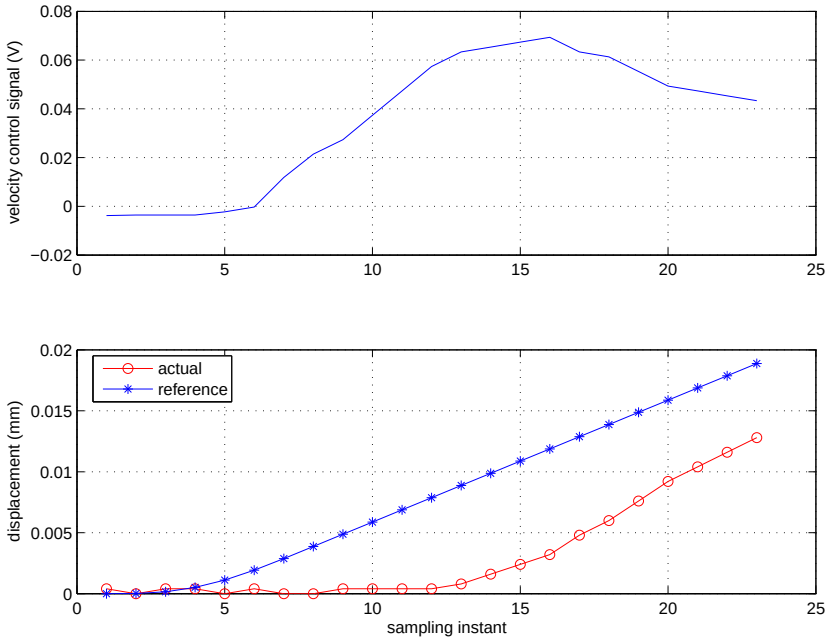

Figure 5.9: Positive u_{max} for Y axis when f=1 mm/s.

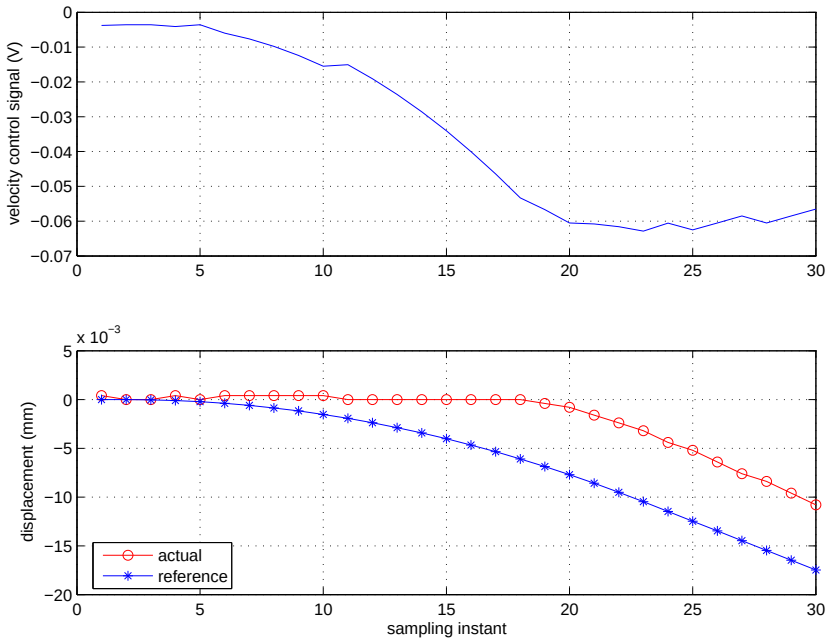

Figure 5.10: Negative u_{max} for X axis when f=1 mm/s.

Table 5.1: Maximum compensation signal u_{max} for positive and negative directions.

feedrate (mm/s)		0.5	1	2	4	8	mean
X	voltage + (V)	0.049504	0.068666	0.058479	0.076326	0.058479	0.062291
	instant (ms)	13	11	9	10	9	
	voltage − (V)	−0.033286	−0.057458	−0.065004	−0.047272	−0.047272	−0.050058
	instant (ms)	12	11	10	9	9	
Y	voltage + (V)	0.053215	0.057340	0.084173	0.064986	0.085173	0.068977
	instant (ms)	16	11	11	10	11	
	voltage − (V)	−0.045415	−0.068540	−0.095373	−0.076186	−0.076186	−0.072340
	instant (ms)	13	11	11	10	10	

a feedrate of between 0.5 mm/s to 8 mm/s were used. The values of the velocity control signal and the axial displacement from standstill were recorded for the initial motion up to 30 sampling periods, or 30 msec.

The experimental results are shown in Fig. 5.7 to 5.10 for both the two axes and in both directions. It can be seen from the figures that the axes remained stationary initially when the velocity control signals are small. They started to move only when the velocity control signal exceed certain values.

The maximum compensation signal u_{max} for any axis in a given direction is taken as the velocity control signal when that axis begins to move. In other words, the value of u_{max} occurs when the sampled values of the displacement of the axis satisfy

$$|d(k)| < |d(k+1)| < |d(k+2)| \tag{5.4.1}$$

Table 5.1 shows the results obtained for the values of u_{max} measured at various feedrates. It is noted that there are some variations to the values of u_{max} obtained and, in the compensation scheme, the average of these values was used.

5.5 Displacement-based stiction compensation

5.5.1 Design of displacement-based stiction compensation

We first attempted a displacement-based two-stage stiction compensation scheme with the compensation-coefficient as shown in Fig. 5.11. The first stage, corresponding to the presliding regime, is displacement dependant and shown in Fig. 5.11 (a). The second segment, corresponding to the Stribeck regime, is velocity dependent and shown in Fig. 5.11 (b).

The displacement from the point of zero velocity d is used as the criterion to switch from the first presliding phase to the second sliding phase when d becomes larger than the breakaway displacement d_b. Referring to Fig. 5.11 (a), the dimensionless displacement is defined as

$$d_o^* = \frac{d}{d_b} \tag{5.5.1}$$

The compensation coefficient for the first segment is given by

$$k_f = d_o^* \tag{5.5.2}$$

For the second phase, the coefficient is given by

$$k_f = \frac{1}{1 + v^*} \tag{5.5.3}$$

where the dimensionless velocity v^* is defined as

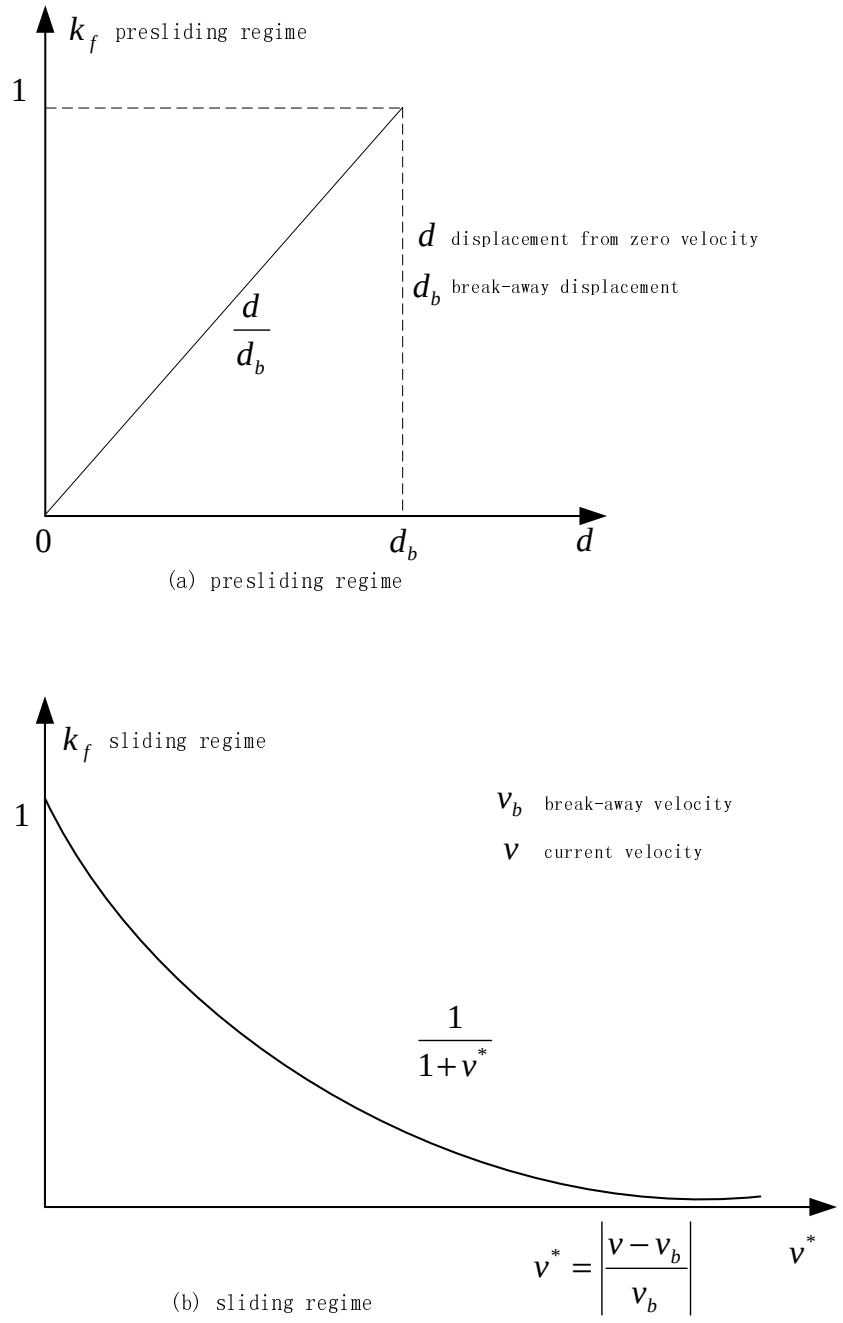

Figure 5.11: Displacement-based two segment compensation coefficient curve: (a) first segment displacement dependent and (b) second segment velocity dependent.

$$v^* = \left| \frac{v - v_b}{v_b} \right| \tag{5.5.4}$$

v_b being the breakaway velocity or the value of the reference velocity recorded at the very first instant when d is equal to or larger than d_b. Eq. (5.5.4) means that the compensation coefficient depends only on the current velocity v, and does not involve the displacement. The value of v_b is used in the following Stribeck phase to compute the compensation coefficient k_f. To avoid the noise associated with the measurement of the actual velocity, the reference velocity is used instead. During the second Stribeck phase, the static friction component that needs to be compensated will drop from its maximum value to zero.

5.5.2 Experimental results for displacement-based stiction compensation

With the displacement-based compensation, a circular contour of radius of 40 mm was generated at a feedrate of 100 mm/s and the results are shown in Fig. 5.12. From the figure, it can be seen that, the applied compensation has a significant effect on the contour errors at the quadrant positions where the error spikes occur, as compared to the uncompensated errors shown in Fig. 5.6 for the same radius and velocity.

It is noted that the contour error spikes at the quadrant positions has been narrowed very significantly, indicating that, beyond these narrow regimes, the compensation scheme has very effectively compensated for stiction effect and reduced the contour errors. However, it is also noted that there is no significant reduction to the maximum

value of the error spikes at the quadrant positions. Instead, "negative" spikes have now occurred immediately after the "positive" spikes even though these latter spikes have been significantly narrowed.

One possible explanation for the observed effect is that the compensation signal was applied too late and when it was applied, there was some degree of over-compensation, leading to the "negative" error spikes. It is clear that the displacement-based two-stage compensation scheme has not worked well in reducing the maximum value of the contour error at the quadrant positions.

There is clearly a need for the stiction compensation scheme similar in approach to the displacement-based scheme but which can have a measured compensation signal applied earlier.

Referring to Figs. 5.7 to 5.10, it can be noted that when an axis starts to move from standstill, the tracking error, representing the difference between reference position and actual position, will lead the actual displacement. In the next section, a tracking error-based compensation scheme is investigated. Using tracking error, the compensation signal can be applied earlier and this is shown to work well.

5.6 Tracking error-based stiction compensation

5.6.1 Design of tracking error-based stiction compensation

Based on the results obtained with the displacement-based stiction compensation scheme presented in the previous section, it seems logical to substitute displacement

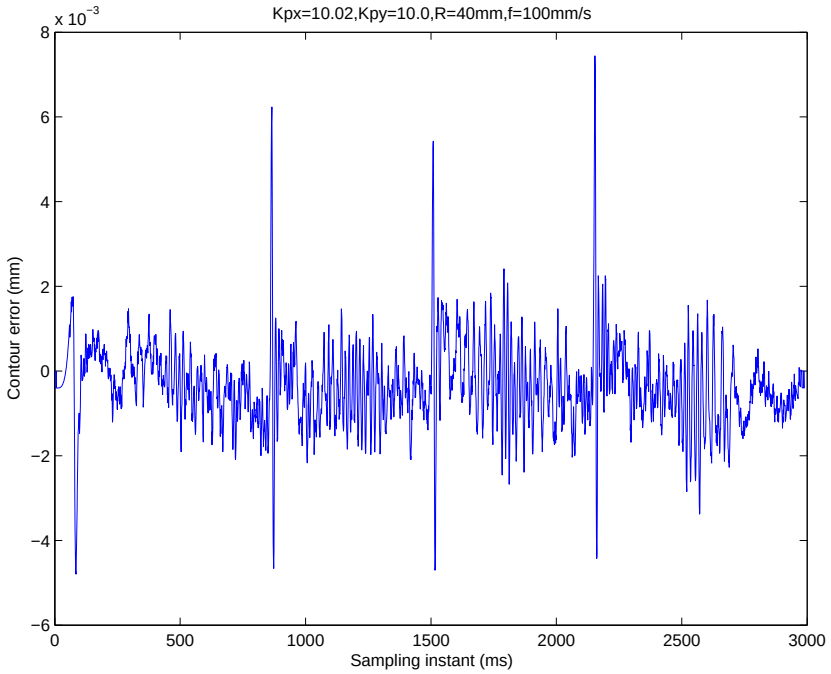

Figure 5.12: Displacement-based stiction compensation when f=100mm/s, R=40mm

with tracking error. Referring to Figs. 5.7 to 5.10, it can be seen that if tracking error e is used to replace displacement d in Eq. (5.5.1), the maximum value of the compensation signal will be applied earlier as the tracking error leads the displacement when the axis moves from standstill. The total effect should thus reduce both the positive as well as the negative error spikes seen in the previous section.

For the tracking error-based compensation, the proposed profile of the compensation coefficient is shown in Fig. 5.13. The first phase, corresponding to the presliding regime, is tracking error dependant and shown in Fig. 5.13 (a). The second phase, corresponding to the Stribeck regime, is velocity dependent and shown in Fig. 5.13 (b). For the continuity of the control signal, the compensation signal in the presliding regime increases linearly from zero to its maximum value of u_{max}. The dimensionless

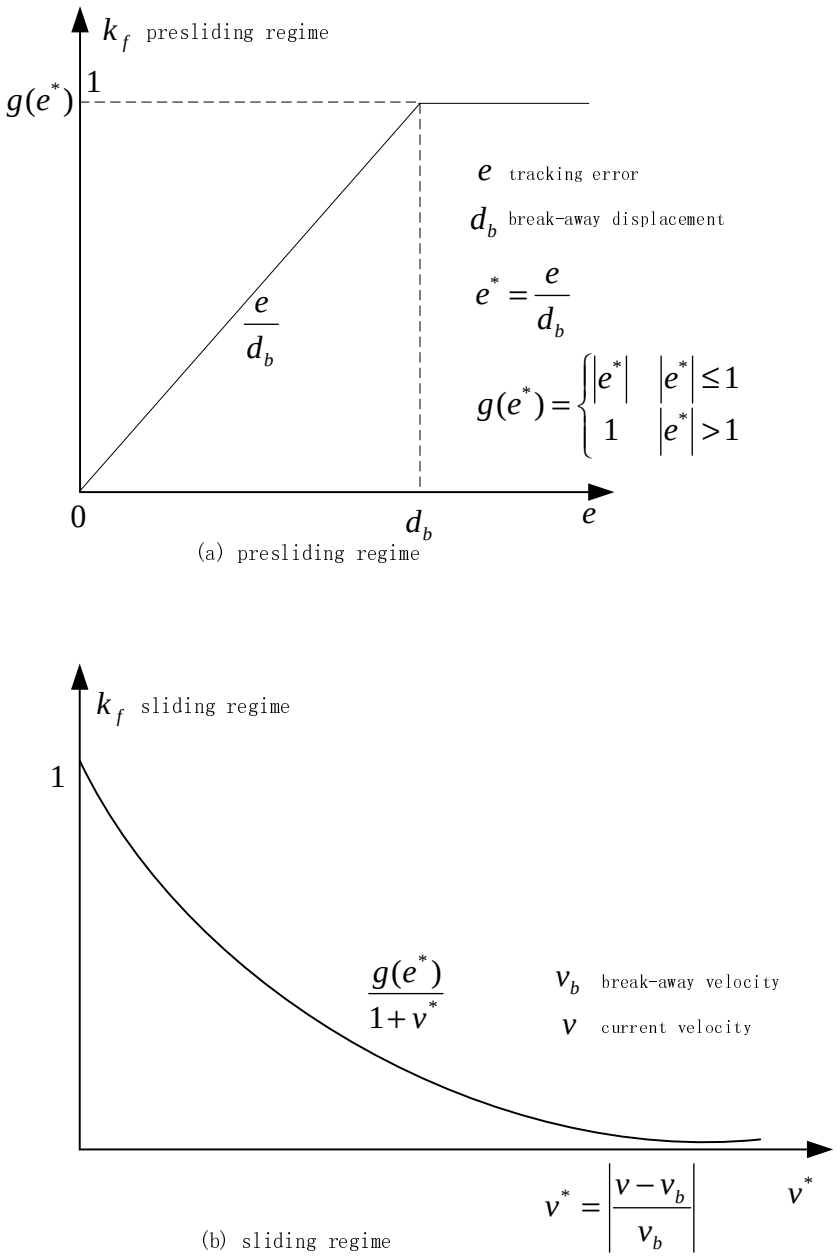

Figure 5.13: Tracking error-based two segment compensation coefficient curve: (a) first segment displacement dependent and (b) second segment velocity dependent.

tracking error e^* is defined as

$$e^* = \frac{e}{d_b} \tag{5.6.1}$$

where e is the tracking error. Referring to Eq. (5.5.1), the displacement from the point of zero velocity, d, is replaced by the tracking error e.

The compensation coefficient used for the first segment is

$$k_f = g(e^*) \tag{5.6.2}$$

It can be noted that, in the first segment, the compensation coefficient k_f is zero when the tracking error e is zero, and increases linearly up to a maximum value of 1.0 when e is equal to d_b.

Referring to Figs. 5.7 to 5.10, together with Fig. 5.13 (a), and noting that the breakaway displacement d_b is between 3−5 μm, it can be seen that the tracking error can become equal to and greater than d_b before the breakaway displacement is reached, or before compensation switches to the second phase when the displacement $d \geq d_b$

A clamping function is thus introduced

$$g(e^*) = \begin{cases} |e^*|, & |e^*| \leq 1 \\ 1, & |e^*| > 1 \end{cases}$$

This is to limit the value of k_f to a maximum of 1, as shown in Fig. 5.13 (a) to avoid over-compensation.

As with the displacement-based scheme, discussed in the preceding section, compensation is switched to the second phase when $d = d_b$ at which point the value of reference velocity is taken as the breakaway velocity, v_b.

With the replacement of d with the tracking error e in Fig. 5.13 (a), there is also a need to modify the profile in Fig. 5.13 (b) to function

$$k_f = \frac{g(e^*)}{1 + v^*} \tag{5.6.3}$$

This is to ensure continuity of k_f when switching form the first phase to the second phase. Whereas in the previous scheme represented by Fig. 5.11, the value of k_f at the point of switching will equal to 1 when $d = d_b$, this may not always hold true in the tracking error-based scheme represented by Fig. 5.13. This is because, under certain operating conditions it may be possible for e to be smaller then d_b when the displacement d increases to d_b at which point the switch is made from the presliding compensation to the sliding compensation. Thus, to ensure continuity of the value of k_f, Eq. (5.6.3) is used. The relationship between k_f and e^* and v^* is shown in Fig. 5.14.

5.6.2 Experimental results for tracking error-based stiction compensation

A series of circular contours were generated to test the performance of the two-stage tracking error-based static friction compensation scheme. With R=40 mm, two circular contours are generated at two different feedrates, 40 mm/s and 100 mm/s.

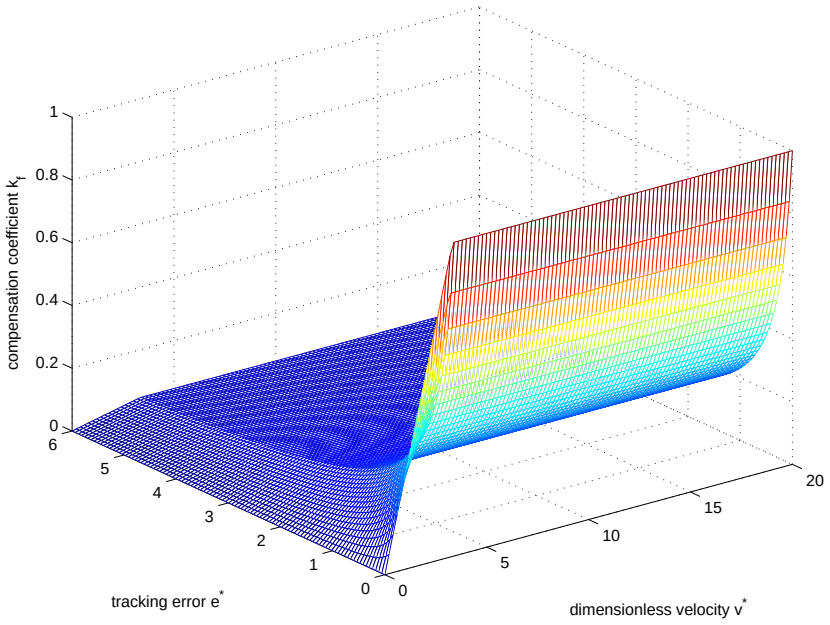

Figure 5.14: Compensation coefficient k_f as a function of dimensionless tracking error e^* and dimensionless velocity v^*

The corresponding contour errors are shown in Figs. 5.15 and 5.16, respectively. These errors can be compared to those without compensation as shown in Figs. 5.5 and 5.6 and to that with displacement-based compensation as shown in Fig. 5.12. For an visual overview of the distribution of the contour errors, the circular contours of these two cases, together with those of the corresponding uncompensated cases, are plotted in the X-Y plane as shown in Figs. 5.17 and 5.18 respectively. The magnitude of the contour errors have been enlarged by 1800 times for better illustration.

It is noted that, without compensation, the contour errors at quadrant positions ranged between 6 and 8 μm because of stiction effect. With tracking error-based stiction compensation, it can be seen from Figs. 5.15 and 5.16 that there is no longer any noticeable increase in contour errors at the quadrant positions, indicating that the

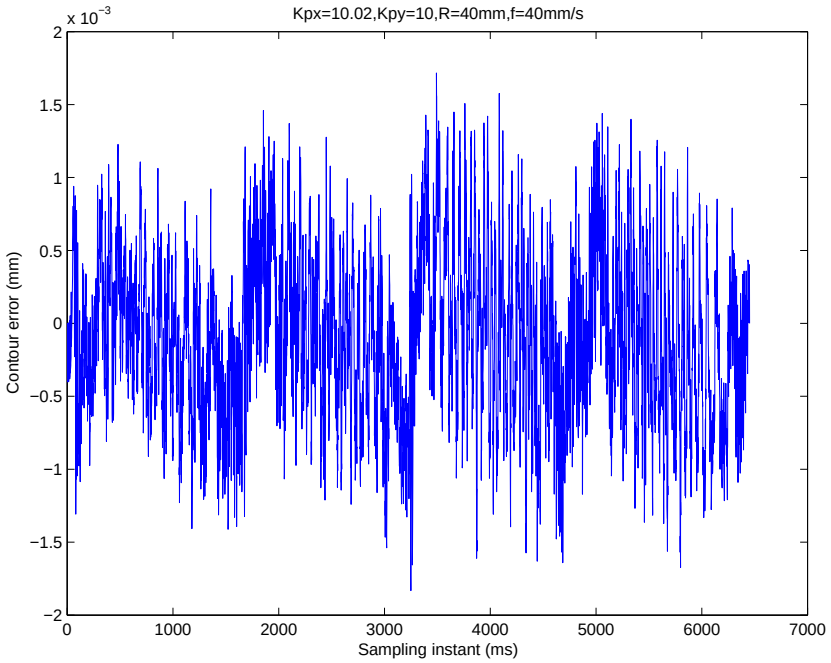

Figure 5.15: Contour error with friction compensation when f=40mm/s, R=40mm

effect of stiction on contour errors have been effectively compensated. What contour errors there remain in the compensated system, in the range of ± 2.5 μm, are caused by vibrations in the motion.

5.7 Conclusion

Contour errors in CNC machines mainly result from unmatched axial dynamics, limited axial bandwidth, and undesirable nonlinearities. Matched axial dynamics can significantly improve the contouring accuracy for both linear and circular contours. For circular contours, the radial errors resulting from the limited bandwidth of the axial dynamics can be further effectively reduced by a simple method of feedforward compensation.

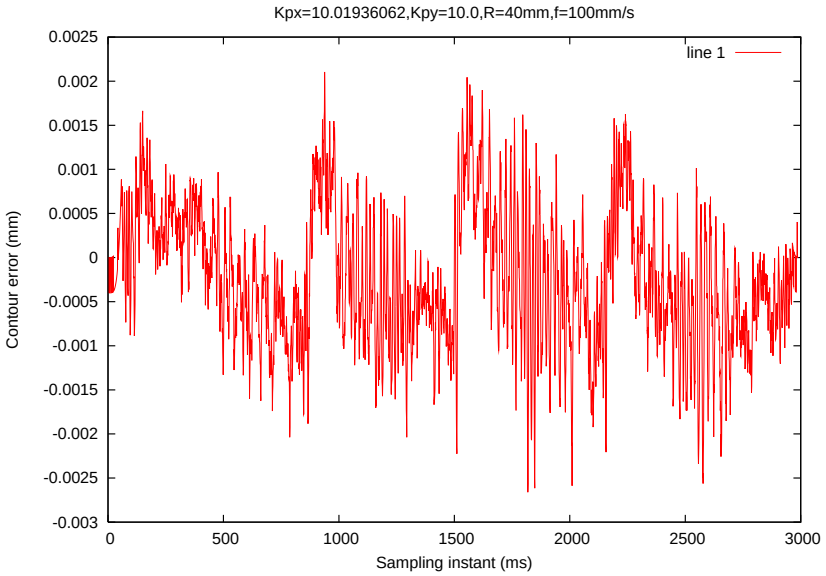

Figure 5.16: Contour error with friction compensation when f=100mm/s, R=40mm

For circular contours, friction, especially static friction, degrades the circular contouring accuracy at quadrant positions when axial velocity changes direction or when moving from standstill. A two-stage static friction compensation scheme is proposed to compensate for static friction induced contour errors. In this two-stage scheme, compensation is divided into two phases.

In the first presliding phase, the compensation signal is a function of tracking error. When motion reaches the breakaway displacement and becomes a sliding motion, the compensation switches to the second phase in which the compensation signal is a function of velocity and tracking error, compensating for the Stribeck effect.

Experimental results shows that, with the two-stage tracking error-based compensation approach, the effect of stiction on contour error can be effectively eliminated. The resulting contour errors are then within values caused by vibrations in the motion of the axes. If the vibration in the motion can be reduced, it can thus be possible to

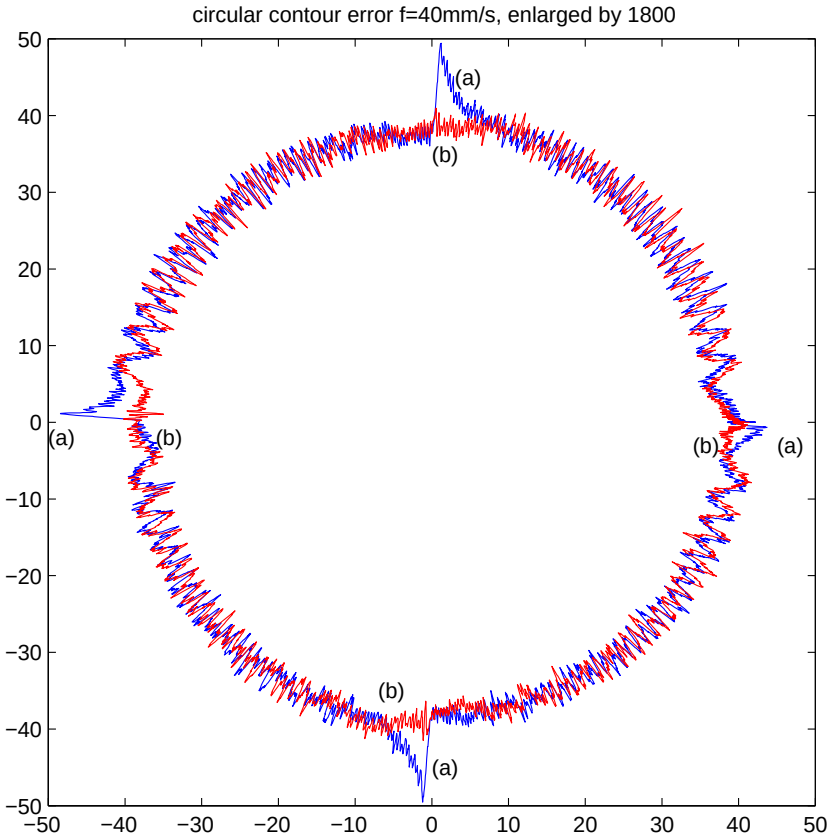

Figure 5.17: X-Y plot of circular contour errors enlarged by 1800 when f=40mm/s, R=40mm: (a) without stiction compensation and (b) with stiction compensation

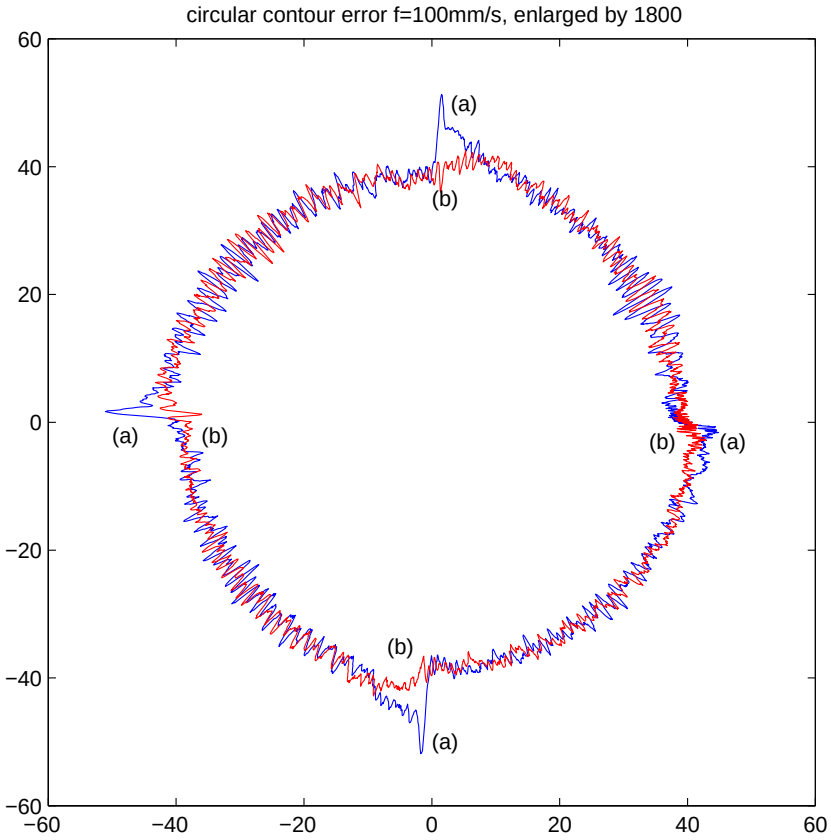

Figure 5.18: X-Y plot of circular contour errors enlarged by 1800 when f=100mm/s, R=40mm: (a) without stiction compensation and (b) with stiction compensation

reduce contour errors even further.

Chapter 6

Taylor Series Expansion Error Compensation

In this chapter, a new model-based Taylor series expansion error compensation (TSEEC) strategy is proposed to improve the contouring accuracy for CNC machines. In TSEEC, the contour error compensation problem is formulated as a Taylor series expansion problem, in which the value of the contour error is expanded around the reference points and the compensation components are calculated as the deviation from the reference points. A comparison of the contouring accuracy performance by both the zero phase error tracking controller (ZPETC), also a model-based approach, and the Taylor series expansion error compensation under the same model error also shows the robustness of the proposed TSEEC error compensation method.

Simulations show that, under conditions of perfect knowledge of the dynamics of the axes, perfect contouring with zero contour errors can be achieved with TSEEC. However, due to modeling errors, external disturbances, and measurement noise, some modifications and experimentation need to be made to determine suitable parameters for implementation of the TSEEC scheme on a real machine. A low-pass filter to

filter out the high-frequency components in the compensation signals which otherwise would cause chattering, needs to be added. The choice of the compensation goal for a time-delayed discrete-time system also needs to be adjusted. Rather than targeting to make the contour error at the $(k+d)$ instant zero, where k is the current sampling instant and d is the time delay in a discrete-time system, this is adjusted to make the moving average of the contour errors in the future d steps zero. A suitable value of the compensation gain which determines the amount of the compensation signal to be used also needs to found from experimentation. This compensation gains can be used to cope with the model errors and other disturbances.

6.1 Introduction

Efforts to reduce or eliminate contour errors are made either through the design of advanced controllers for feed drives [30] or by the method of path precompensation [12, 11, 10]. The design of advanced controllers can be further divided into 1) tracking control and 2) contouring control. The idea of tracking control is to reduce or cancel the tracking or following error of individual axes, thus indirectly reducing the contour error. Tomizuka [56] proposed the zero phase error tracking controller (ZPETC) which can achieve zero phase lag and unity DC gain for reference inputs. While tracking control aims to improve the tracking accuracy of individual axes, the cross-coupled controller (CCC) proposed by Koren [27] is devised to eliminate or reduce contour errors directly, rather than the tracking errors in individual axes. CCC controllers estimate the contour error in real time and utilizes it in a control algorithm that reduces or eliminates the contour error.

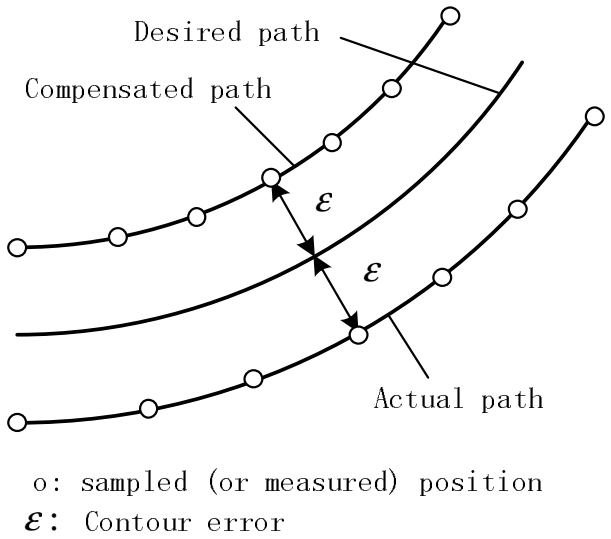

Figure 6.1: A biaxial contour machining results and the compensation strategy [34].

While the aforementioned control methods do improve the contouring accuracy, it may not be easy to implement on existing CNC machines because of the need for modifications to the existing controllers. An alternative way to improve contouring accuracy is by using path precompensation, which is to modify the reference position inputs before feeding these to the individual axial control loops. Chin and Tsai [12] proposed the path precompensation method (PPM), and later introduced a cross-coupled path precompensation (CCPM) which combines the cross-coupling technique and the path precompensation method [11]. In [10], they used fuzzy logic to enhance the performance of CCPM.

However, all the above methods utilize only the information of the past. The corrections are made to the reference position inputs only after the contour error occurs. If an error model can be developed which can predict the contour error, then this could be used to generate the necessary path precompensation and compensate the contour error even before they occur. Some work has been done in this direction but

only for offline applications. Lo and Hsiao [34, 35] proposed a method of tool path compensation for repeated machining of identical parts. In their method, the profile of the first machined part is measured by a coordinate measuring machine. Based on the measured contour error data, a path compensation algorithm is applied to generate a modified tool path which is offset from the original tool path by the contour error as illustrated in Fig. 6.1. This modified tool path is then utilized as the reference input for machining of the subsequent parts. In this approach, the prediction of contour errors is replaced by the use of the recorded contour errors in a previous machining. This works as long as the machining process is repeatable. Despite its simplicity, however, this method cannot compensate completely for the contour error. The required path, which can compensate completely for the contour error is not as simple as the one offset by the contour error as shown in Fig. 6.1 because of the complex relationship between the reference inputs and the resulting contour errors. Their simulation and experimental results confirmed this.

Ye *et al.* [64] proposed a cross-coupled path precompensation, which first utilizes a prediction model to estimate tracking errors, then estimating the contour errors based on these tracking errors. These contour errors are then used to calculate the compensation components along individual axes. However, because of the assumption that the tracking errors remain unchanged before and after compensation, which is not the case as the reference inputs are different in the two cases, this method is still unable to completely compensate for the contour error.

In this chapter, a new contour error compensation strategy, which is based on the Taylor series expansion, is used to improve the contouring accuracy for CNC machines. The axial dynamic models are first used to predict the contour error based on the

reference inputs to be applied. The Taylor series expansion is then used to develop a contour error model around the current position along the contour path. A numerical method is devised which then determines the required modified reference inputs along individual axes so as to compensate for the contour error.

Simulation results show that the proposed approach can completely compensate for the contour errors for both linear and circular contours even with mismatches between axes' dynamics. A comparison of the contouring accuracy performance of the zero phase error tracking controller (ZPETC) and the Taylor series expansion error compensation (TSEEC) in the presence of modeling errors also shows the robustness and superior performance of the proposed error compensation method.

However, due to modeling errors, external disturbances, and measurement noise, some modifications and experimentation need to be made to determine suitable parameters for implementation of the TSEEC scheme on a real machine. These include a low-pass filter, a modified compensation target and compensation gains. Experimental results show that TSEEC can effectively reduce contour errors.

The subsequent sections are arranged as follows: Section 6.2 introduces the formulation for TSEEC, and Section 6.3 illustrates by simulation the contouring performance using the proposed error compensation strategy. A comparison with ZPETC is carried out in Section 6.4. Section 6.5 presents some necessary modifications and the experimental results of the implementation of TSEEC on a mini-CNC. Finally Section 6.6 presents some discussions and a conclusion.

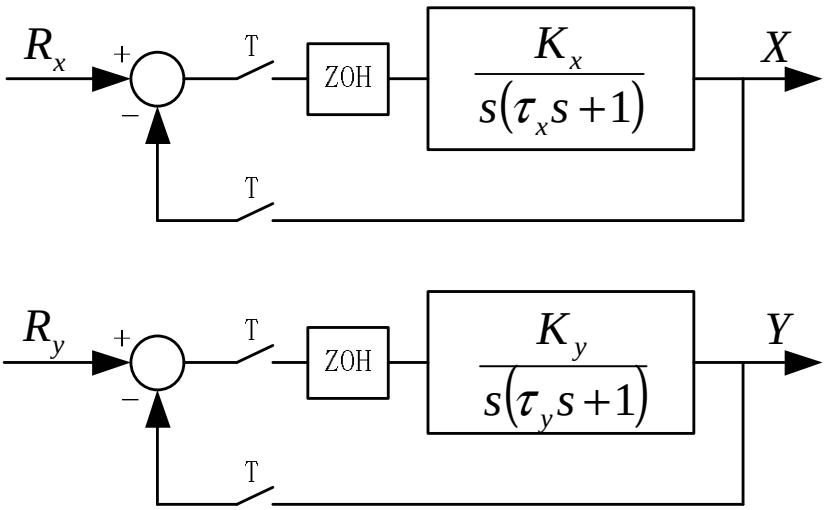

Figure 6.2: System model for bi-axis contouring control system.

6.2 Error Compensation Based on Taylor Series Expansion

Poo *et al.* [43] have shown that the mismatches between axes dynamics result in contour errors. When a desired circle is machined, the actual path is like an ellipse with the magnitude and orientation of the contour error depending on the degree of mismatched dynamics and the feedrate. The proposed compensation strategy is based on the Taylor series expansion.

A two-axis contouring control system is shown in Fig. 6.2 comprising two sampled-data position feedback control systems, one for the X-axis and the other for the Y-axis. The open-loop transfer function of the two axes, X and Y, are of second order and given by

$$G_{ox}(s) = \frac{K_x}{s(\tau_x s + 1)}, \quad G_{oy}(s) = \frac{K_y}{s(\tau_y s + 1)} \tag{6.2.1}$$

where K_x and K_y are gain constants and τ_x and τ_y are time constants. With a zero order hold preceding the servomotor and the sampling interval chosen as T, the closed-loop discrete-time models are given by

$$H_x(q^{-1}) = \frac{b_{11}q^{-1} + b_{12}q^{-2}}{1 + a_{11}q^{-1} + a_{12}q^{-2}} \tag{6.2.2}$$

$$H_y(q^{-1}) = \frac{b_{21}q^{-1} + b_{22}q^{-2}}{1 + a_{21}q^{-1} + a_{22}q^{-2}} \tag{6.2.3}$$

where a_{ij} and b_{ij} are constants, and q^{-1} is the unit time-delay operator. Denoting the current sampling instant as k, the position at the $(k+1)^{th}$ instant can be derived, using Eq. (6.2.2) and (6.2.3), as

$$x(k+1) = -a_{11}x(k) - a_{12}x(k-1) + b_{11}r_x(k) + b_{12}r_x(k-1) \tag{6.2.4}$$

$$y(k+1) = -a_{21}y(k) - a_{22}y(k-1) + b_{21}r_y(k) + b_{22}r_y(k-1) \tag{6.2.5}$$

where x and y are the actual X and Y coordinates of the resulting motion respectively, and r_x and r_y are the actual inputs to the X- and the Y-axis respectively.

Let r_x^* and r_y^* be the reference inputs without application of any compensation. These are related by

$$r_x(k) = r_x^*(k) + \Delta r_x(k), \qquad r_y(k) = r_y^*(k) + \Delta r_y(k) \tag{6.2.6}$$

where $\Delta r_x(k)$ and $\Delta r_y(k)$ are the respective correction or compensation terms.

From Eqs. (6.2.4) and (6.2.5), it can be noted that as long as the actual inputs are known, the response, and thus, the contour error, can be computed. The problem is then to determine a set of inputs, based on these reference inputs, so as to achieve

zero contour error. For an arbitrary curve, the compensated contour error at the $(k+1)^{th}$ instant can be expressed as

$$\begin{aligned} \varepsilon(k+1) &= f(x(k+1), y(k+1)) \\ &= f(x(k), x(k-1), r_x(k), r_x(k-1), y(k), y(k-1), r_y(k), r_y(k-1)) \end{aligned} \tag{6.2.7}$$

Noting the fact that at the k^{th} instant all the parameters in the function $f(\cdot)$ are constants except for the actual inputs $r_x(k)$ and $r_y(k)$, Eq. (6.2.7) can be rewritten as

$$\varepsilon(k+1) = f(r_x(k), r_y(k)) = f(r_x^*(k) + \Delta r_x(k), r_y^*(k) + \Delta r_y(k)) \tag{6.2.8}$$

If no compensation is made at the k^{th} instant, i.e. $\Delta r_x(k) = \Delta r_y(k) = 0$, then the contour error will be

$$\varepsilon^*(k+1) = f(r_x^*(k), r_y^*(k)) \tag{6.2.9}$$

Using the Taylor series expansion and expanding about the point $\big(r_x^*(k), r_y^*(k)\big)$, neglecting terms with order higher than one, and forcing the resulting contour error to zero, Eq. (6.2.8) can be written as

$$\varepsilon(k+1) \doteq f(r_x^*(k), r_y^*(k)) + \frac{\partial f}{\partial r_x(k)} \Delta r_x(k) + \frac{\partial f}{\partial r_y(k)} \Delta r_y(k) = 0 \tag{6.2.10}$$

For convenience, denote the two partial derivatives as

$$\frac{\partial f(r_x(k), r_y(k))}{\partial r_x(k)}\Big|_{r_x(k)=r_x^*(k), r_y(k)=r_y^*(k)} = A \tag{6.2.11}$$

$$\frac{\partial f(r_x(k), r_y(k))}{\partial r_y(k)}\Big|_{r_x(k)=r_x^*(k), r_y(k)=r_y^*(k)} = B \tag{6.2.12}$$

Substituting Eq. (6.2.9), (6.2.11) and (6.2.12) into Eq. (6.2.10) gives

$$-\varepsilon^*(k+1) = A\Delta r_x(k) + B\Delta r_y(k) \tag{6.2.13}$$

The determination of the error compensation terms then reduces to obtaining the values of $\Delta r_x(k)$ and $\Delta r_y(k)$ to satisfy Eq. (6.2.13). Note that there are infinite number of possible sets of solutions for Eq. (6.2.13). As such, an additional auxiliary equation is needed to determine the relationship between $\Delta r_x(k)$ and $\Delta r_y(k)$. This has a significant effect on the robustness of the compensation algorithm and, thus, the contouring accuracy. Different auxiliary equations are devised for linear and circular contours and these will be discussed in the following sections.

6.2.1 Linear Contours

Consider first the linear contour. Let the desired linear contour be at an angle θ with respect to the X-axis. Referring to Fig. 4.2, the contour error at the $(k+1)^{th}$ instant is given by

$$\varepsilon(k+1) = e_y(k+1)\cos\theta - e_x(k+1)\sin\theta \tag{6.2.14}$$

where e_x and e_y are the tracking errors along the X-axis and the Y-axis respectively. Also, using Eq. (6.2.4) and (6.2.5), we have

$$\begin{aligned} e_x(k+1) &= r_x^*(k+1) - x(k+1) \\ &= r_x^*(k+1) - (-a_{11}x(k) - a_{12}x(k-1) + b_{11}r_x(k) + b_{12}r_x(k-1)) \end{aligned} \tag{6.2.15}$$

$$\begin{aligned} e_y(k+1) &= r_y^*(k+1) - y(k+1) \\ &= r_y^*(k+1) - (-a_{21}y(k) - a_{22}y(k-1) + b_{21}r_y(k) + b_{22}r_y(k-1)) \end{aligned} \tag{6.2.16}$$

Substituting Eq. (6.2.15) and (6.2.16) into Eq. (6.2.14), we have

$$\begin{aligned} \varepsilon(k+1) &= r_y^*(k+1)\cos\theta - r_x^*(k+1)\sin\theta \\ &\quad +(-a_{11}x(k) - a_{12}x(k-1) + b_{11}r_x(k) + b_{12}r_x(k-1))\sin\theta \\ &\quad -(-a_{21}y(k) - a_{22}y(k-1) + b_{21}r_y(k) + b_{22}r_y(k-1))\cos\theta \end{aligned} \tag{6.2.17}$$

Since

$$r_x^*(k+1) = r^*(k+1)\cos\theta, \qquad r_y^*(k+1) = r^*(k+1)\sin\theta \tag{6.2.18}$$

Eq. (6.2.17) then becomes

$$\begin{aligned} \varepsilon(k+1) &= f(r_x(k), r_y(k)) \\ &= \sin\theta(-a_{11}x(k) - a_{12}x(k-1) + b_{11}r_x(k) + b_{12}r_x(k-1)) \\ &\quad -\cos\theta(-a_{21}y(k) - a_{22}y(k-1) + b_{21}r_y(k) + b_{22}r_y(k-1)) \end{aligned} \tag{6.2.19}$$

The derivative of contour error with respect to $r_x(k)$ and $r_y(k)$ is deduced from Eq. (6.2.19)

$$\frac{\partial f}{\partial r_x(k)} = b_{11}\sin\theta\Big|_{r(x)=r_x^*(k),r(y)=r_y^*(k)} = A \tag{6.2.20}$$

$$\frac{\partial f}{\partial r_y(k)} = -b_{21}\cos\theta\Big|_{r(x)=r_x^*(k),r(y)=r_y^*(k)} = B \tag{6.2.21}$$

The contour error without compensation, $\varepsilon^*(k+1)$, is simply obtained by Eq. (6.2.19) when $r_x(k) = r_x^*(k)$ and $r_y(k) = r_y^*(k)$. Considering Eq. (6.2.13), another equation is needed to calculated the compensation components along the individual axes. For computational robustness, this equation is devised as

$$\frac{\cos\theta}{b_{21}}\Delta r_x(k) + \frac{\sin\theta}{b_{11}}\Delta r_y(k) = 0 \tag{6.2.22}$$

The compensation components are then given by

$$\begin{pmatrix} b_{11}\sin\theta & -b_{21}\cos\theta \\ \frac{\cos\theta}{b_{21}} & \frac{\sin\theta}{b_{11}} \end{pmatrix}\begin{pmatrix} \Delta r_x(k) \\ \Delta r_y(k) \end{pmatrix} = \begin{pmatrix} -\varepsilon(k+1) \\ 0 \end{pmatrix} \tag{6.2.23}$$

Finally, the compensation components are determined as

$$\begin{pmatrix} \Delta r_x(k) \\ \Delta r_y(k) \end{pmatrix} = \begin{pmatrix} \frac{\sin\theta}{b_{11}} & b_{21}\cos\theta \\ -\frac{\cos\theta}{b_{21}} & b_{11}\sin\theta \end{pmatrix}\begin{pmatrix} -\varepsilon(k+1) \\ 0 \end{pmatrix} = \begin{pmatrix} -\frac{A\varepsilon(k+1)}{b_{11}^2} \\ \frac{B\varepsilon(k+1)}{b_{21}^2} \end{pmatrix} \tag{6.2.24}$$

6.2.2 Circular Contours

Consider next the circular contour. The contour error at the $(k+1)^{th}$ instant is

$$\varepsilon(k+1) = f(r_x(k), r_y(k)) \tag{6.2.25}$$

$$= \sqrt{x(k+1)^2 + y(k+1)^2} - R$$

$$= \sqrt{(-a_{11}x(k) - a_{12}x(k-1) + b_{11}r_x(k) + b_{12}r_x(k-1))^2 + (-a_{21}y(k) - a_{22}y(k-1) + b_{21}r_y(k) + b_{22}r_y(k-1))^2} - R$$

As with the linear contour, the contour error without compensation, $\varepsilon^*(k+1)$, is obtained with $r_x(k) = r_x^*(k)$ and $r_y(k) = r_y^*(k)$. The derivative of this contour error with respect to $r_x(k)$ and $r_y(k)$ is deduced from Eq. (6.2.25) as

$$\frac{\partial f}{\partial r_x(k)}\Big|_{r_x(k)=r_x^*(k), r_y(k)=r_y^*(k)} = \frac{b_{11}x(k+1)}{\sqrt{x(k+1)^2+y(k+1)^2}} = A \qquad (6.2.26)$$

$$\frac{\partial f}{\partial r_y(k)}\Big|_{r_x(k)=r_x^*(k), r_y(k)=r_y^*(k)} = \frac{b_{21}y(k+1)}{\sqrt{x(k+1)^2+y(k+1)^2}} = B \qquad (6.2.27)$$

For the circular contour, two different auxiliary equations are devised. The first scheme allocates the compensation components proportional to their respective derivatives

$$\frac{\Delta r_x(k)}{\Delta r_y(k)} = \frac{A}{B} \qquad (6.2.28)$$

for which $\Delta r_x(k)$ and $\Delta r_y(k)$ become

$$\begin{pmatrix} A & B \\ -B & A \end{pmatrix} \begin{pmatrix} \Delta r_x(k) \\ \Delta r_y(k) \end{pmatrix} = \begin{pmatrix} -\varepsilon(k+1) \\ 0 \end{pmatrix} \qquad (6.2.29)$$

In this case,

$$A^2 + B^2 = \frac{b_{12}^2 x(k+1)^2 + b_{22}^2 y(k+1)^2}{x(k+1)^2 + y(k+1)^2} > 0 \qquad (6.2.30)$$

and this scheme is numerically robust. The second scheme is also devised as a numerically robust algorithm to calculate the compensation component online. As will be shown in the next section, the second scheme is more effective in improving contouring accuracy as compared with the first. Noting the fact that $b_{11} > 0$ and $b_{21} > 0$, and also noting that

$$\frac{A^2}{b_{11}^2} + \frac{B^2}{b_{21}^2} = \frac{x(k+1)^2}{x(k+1)^2 + y(k+1)^2} + \frac{y(k+1)^2}{x(k+1)^2 + y(k+1)^2} = 1 \tag{6.2.31}$$

The auxiliary equation is devised as

$$-\frac{B}{b_{21}^2}\Delta r_x(k) + \frac{A}{b_{11}^2}\Delta r_y(k) = 0 \tag{6.2.32}$$

This gives

$$\begin{pmatrix} A & B \\ -\frac{B}{b_{21}^2} & \frac{A}{b_{11}^2} \end{pmatrix} \begin{pmatrix} \Delta r_x(k) \\ \Delta r_y(k) \end{pmatrix} = \begin{pmatrix} -\varepsilon(k+1) \\ 0 \end{pmatrix} \tag{6.2.33}$$

The compensation components along the individual axes are then given by

$$\begin{pmatrix} \Delta r_x(k) \\ \Delta r_y(k) \end{pmatrix} = \begin{pmatrix} A & B \\ -\frac{B}{b_{21}^2} & \frac{A}{b_{11}^2} \end{pmatrix}^{-1} \begin{pmatrix} -\varepsilon(k+1) \\ 0 \end{pmatrix} = \begin{pmatrix} -\frac{A\varepsilon(k+1)}{b_{11}^2} \\ -\frac{B\varepsilon(k+1)}{b_{21}^2} \end{pmatrix} \tag{6.2.34}$$

It is noted that the compensation components for linear contours have similar expressions to the case of circular contours except that the partial derivatives, A and B, and the signs are different.

6.3 Simulation Study

In this section, the proposed Taylor series expansion error compensation strategy is applied to compensate for both linear and circular contours for a test system with time constants of $\tau_x = 0.01$ sec and $\tau_y = 0.015$ sec. The gains, K_x and K_y, are chosen such that the damping ratios for both axes are 0.707. The sampling interval T is chosen as 0.001 sec. The desired linear contour is at an angle of 30° with respect to the X-axis and the feedrate $f = 1$. Assuming the starting point is at $(0, 0)$, the reference position inputs for the linear contour are given by

$$L = fTk, \tag{6.3.1a}$$

$$r_x(k) = L\cos\theta, \tag{6.3.1b}$$

$$r_y(k) = L\sin\theta \tag{6.3.1c}$$

where L is the distance between $(r_x(k), r_y(k))$ and $(0,0)$, f is the feedrate, T is the sampling interval, k is the current sampling instant and θ is the angle of the linear contour with respect to X-axis. For the circular contour, the desired radius $R = 1$ and the feedrate $f = 1$. It is assumed that the origin is at $(0,0)$ and the starting point is at $(R,0)$. The reference position inputs for the circular contour are then given by

$$\theta = \frac{fTk}{R} \tag{6.3.2a}$$

$$r_x(k) = R\cos\theta, \tag{6.3.2b}$$

$$r_y(k) = R\sin\theta \tag{6.3.2c}$$

where θ denotes the angle of the arc from the starting point.

The uncompensated contour errors for both the linear and the circular contours are shown Fig. 6.3. Because of the mismatch between the two axes, the actual steady-state linear contour is offset from the desired path by a constant amount. For the circular contour, the actual contour is approximately elliptical.

The Taylor series expansion error compensation strategy is then used to improve the contouring accuracy. For the linear contour, Eq. (6.2.24) is used to generate the compensation components which are added to the reference position inputs, and the result is shown in Fig. 6.4. It can be seen that the contour errors have been effectively

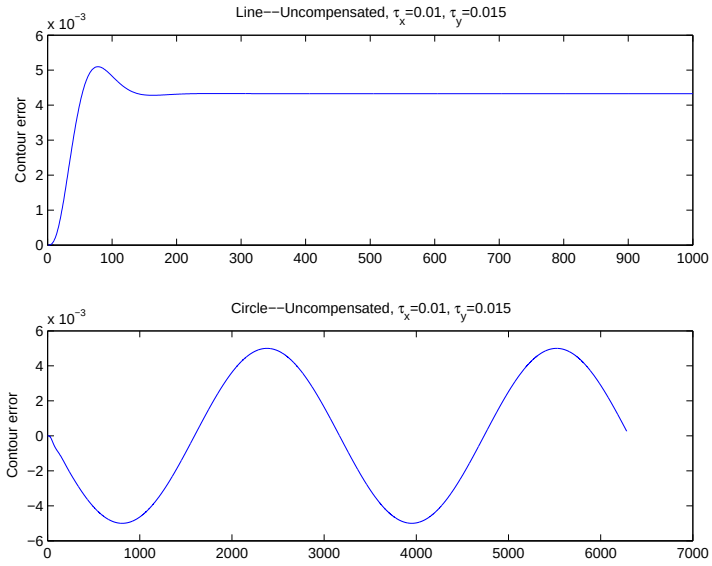

Figure 6.3: Uncompensated contour errors for linear and circular contours.

eliminated with the magnitude in the order of 10^{-15}. The contour errors can thus be considered to have been canceled out completely. The compensation components, Δr_x and Δr_y, shown in Fig. 6.5 indicates that after the transient period, the steady-state compensation components have constant values.

For the circular contour, two schemes were applied. The first scheme allocates the compensation components proportional to their respective derivatives, i.e., $\Delta r_x / \Delta r_y = A/B$. The result is shown in Fig. 6.6. It can be seen that the magnitude of the contour errors is in the order of 10^{-11}. For the second scheme, using Eq. (6.2.34), the result is shown in Fig. 6.7. Here it can be seen that the magnitude of the contour errors is now of the order of 10^{-16} and the contour errors can be considered to have been effectively eliminated. Not only have the steady-state errors been removed, the contour errors during the transient period have also been canceled out. While most of

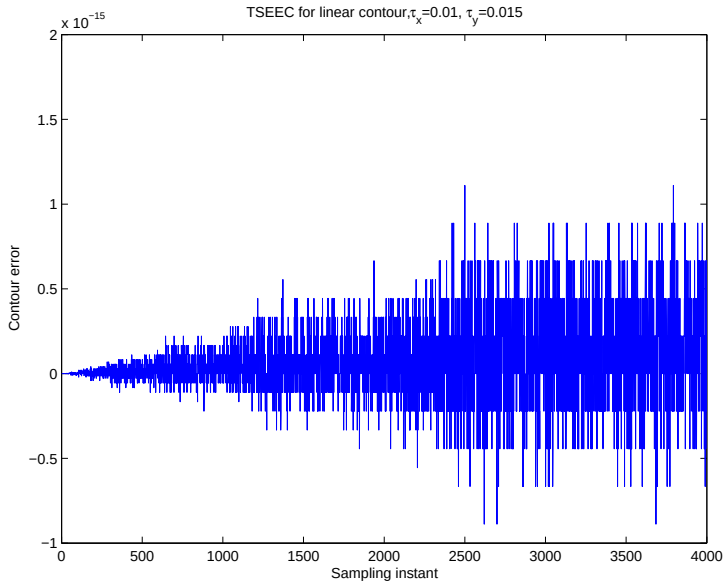

Figure 6.4: contour error for linear contour.

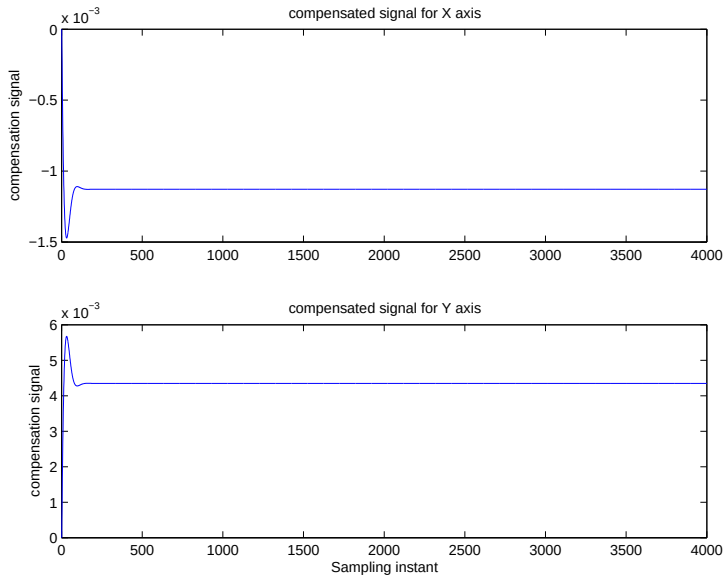

Figure 6.5: compensation signals Δr_x and Δr_y for linear contour.

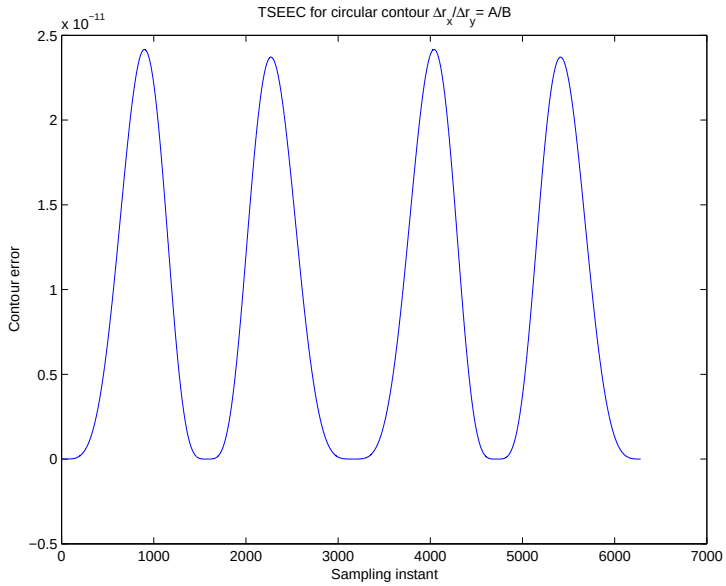

Figure 6.6: Taylor series expansion compensation scheme 1 for circular contour.

the existing contour accuracy improvement strategies only focus on the steady-state contour errors, the method proposed in this chapter can effectively also eliminate the contour errors during the transient phase.

As the servo motor is basically a low-pass filter, high-frequency compensation signals will not be effective. The corresponding compensation components Δr_x and Δr_y of the two schemes for circular contours are shown in Figs. 6.8 and 6.9, respectively. It is noticed that the compensation signals do not contain high-frequency components. Therefore the low frequency compensated trajectory can be practically tracked by the servo loop.

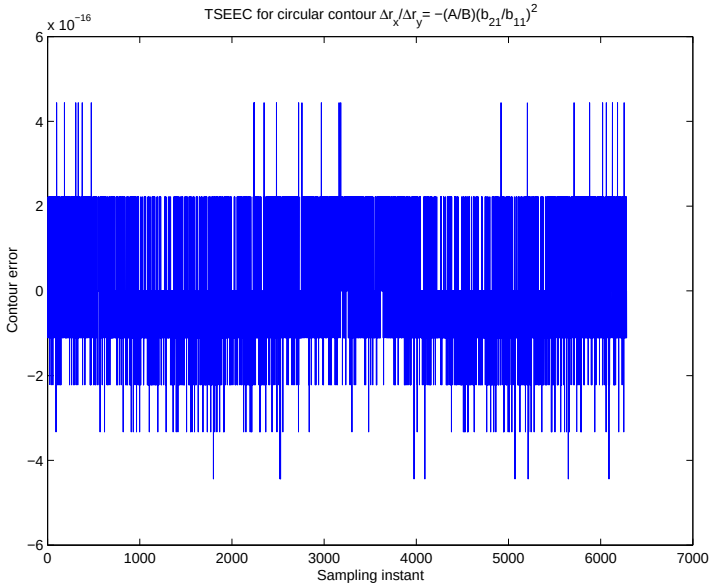

Figure 6.7: Taylor series expansion compensation scheme 2 for circular contour.

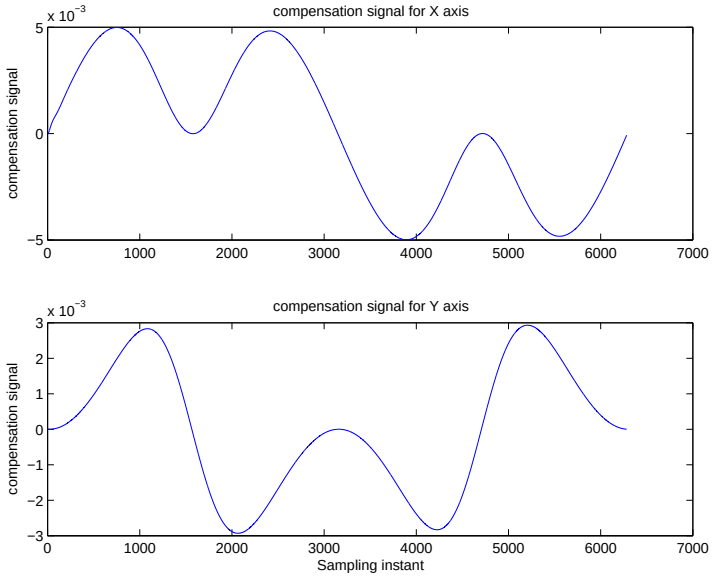

Figure 6.8: compensation signals Δr_x and Δr_y for circular contour(Scheme 1).

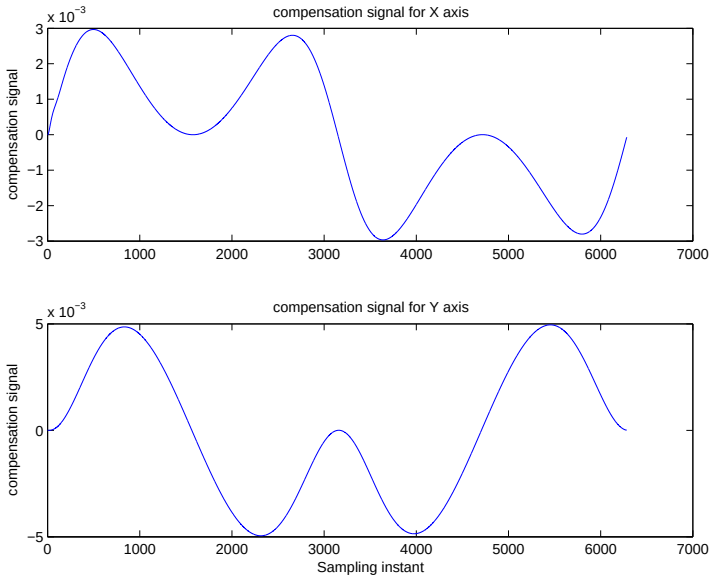

Figure 6.9: compensation signals Δr_x and Δr_y for circular contour(Scheme 2).

6.4 Comparison with feedforward controller

For complete elimination of the contour errors, the proposed error compensation strategy requires a precise knowledge of the dynamics of the system. Feedforward controllers, including ZPETC, also have a similar requirement for good performance. The performance of feedforward controllers depends on the accuracy of the plant model used. In this section, we compare the contouring performance of the ZPETC feedforward controller and the proposed error compensation strategy under conditions of modeling error.

6.4.1 Design of ZPETC controller

The design of the zero phase error tracking controller (ZPETC) was first proposed by Tomizuka [56]. Suppose the closed-loop discrete-time transfer function of the axial dynamics, which includes the controlled plant and feedback controller, is expressed as:

$$G_c(q^{-1}) = \frac{q^{-d}B^+(q^{-1})B^-(q^{-1})}{A(q^{-1})} \tag{6.4.1}$$

where q^{-d} represents a delay of steps of the axis position feedback loop, $A(q^{-1})$ includes the closed-loop poles, $B^+(q^{-1})$ includes the acceptable closed-loop zeros, and $B^-(q^{-1})$ includes the unacceptable closed-loop zeros. The "acceptable" zeros refer to the zeros inside the unit circle and can be taken as the poles in the feedforward controller. By contrast, unacceptable zeros are located outside the unit circle and cannot be poles of the feedforward controller. In practice, some stable zeros which lie on the negative real axis but are very close to -1 (e.g. -0.97) would also be considered as unacceptable zeros for they may cause undesirable oscillations [30].

If unacceptable zeros exist, the feedforward controller cannot be implemented as the inverse of the plant. For such cases, Tomizuka [56] modified the feedforward controller structure and proposed the ZPETC which can achieve zero phase lag and unity DC gain for the reference inputs. Based on Eq. (6.4.1), a ZPETC is designed as

$$G_f(q) = \frac{q^d A(q^{-1})B^-(q)}{B^+(q^{-1})[B^-(1)]^2} \tag{6.4.2}$$

For the test system discussed earlier in Section 6.2, it is found from the above analysis that the zeros of the closed-loop transfer functions of the two axial dynamics are on

the negative real axis and very close to -1. For the X-axis, $z = -0.9672$, and for the Y-axis, $z = -0.9780$. Both zeros are considered as unacceptable zeros. Therefore the ZPETC is chosen rather than an inverse controller which is based on pole/zero cancelation. Comparing Eq. (6.2.2), (6.2.3) and (6.4.1), we have

$$\begin{cases} A_i(q^{-1}) & = 1 + a_{i1}q^{-1} + a_{i2}q^{-2} \quad , i = x, y \\ B_i^+(q^{-1}) & = 1 \\ B_i^-(q^{-1}) & = b_{i1} + b_{i2}q^{-1} \end{cases} \tag{6.4.3}$$

with $d_i = 1$, $i = x, y$. Thus the ZPETC, i.e. the transfer function from $r_i(k)$ to $r_{fi}(k)$, is

$$\frac{R_{fi}(q)}{R_i(q)} = \frac{qA_i(q^{-1})B_i^-(q)}{[B_i^{-1}(1)]^2} = \frac{q(1 + a_{i1}q^{-1} + a_{i2}q^{-2})(b_{i1} + b_{i2}q)}{(b_{i1} + b_{i2})^2} \tag{6.4.4}$$

where $r_i(k)$ and $r_{fi}(k)$ are the reference input and the filtered reference input, respectively. The ZPETC can thus be explicitly written as follows:

$$r_{fi}(k) = \frac{1}{(b_{i1} + b_{i2})^2}\left[b_{i2}r_i(k+2) + (b_{i1} + a_{i1}b_{i2})r_i(k+1) + (a_{i1}b_{i1} + a_{i2}b_{i2}) + a_{i2}b_{i1}r_i(k-1)\right] \tag{6.4.5}$$

6.4.2 Contouring accuracy for circular and linear contours with model error

In this section, the effect of modeling error, or imperfect knowledge of the plant, is considered.

A coefficient of mismatch is defined as:

$$\gamma_i = \frac{\hat{\tau}_i}{\tau_i}, \qquad i = x, y \tag{6.4.6}$$

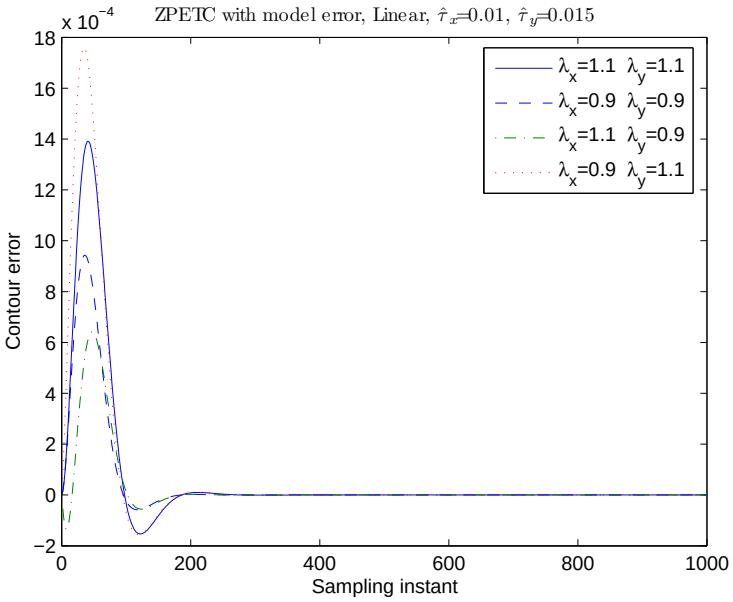

Figure 6.10: Linear contour error by ZPETC with ±10% estimation error in time constant.

where τ_i and $\hat{\tau}_i$ are the actual and the estimated time constants, respectively, of the plant. As done in Section 6.3, the gains K_i are adjusted based on the estimated values of $\hat{\tau}_i$ such that the damping ratio of the nominal closed-loop system is 0.707.

Consider the cases where the mismatches are about 10%, or the following four cases: 1) $\lambda_x = 1.1, \lambda_y = 1.1$; 2) $\lambda_x = 0.9, \lambda_y = 0.9$; 3) $\lambda_x = 1.1, \lambda_y = 0.9$ and 4) $\lambda_x = 0.9, \lambda_y = 1.1$. For these four cases, simulations were done to compare the performance of both the ZPETC and the TSEEC methods in compensating for contour errors.

For each of the four cases mentioned, ZPETC and TSEEC were applied for linear contours. The results are shown in Figs. 6.10 and 6.11. From the results, it is clear that the magnitude of the contour errors when using TSEEC is 3 orders of magnitude smaller than that when using the ZPETC method.

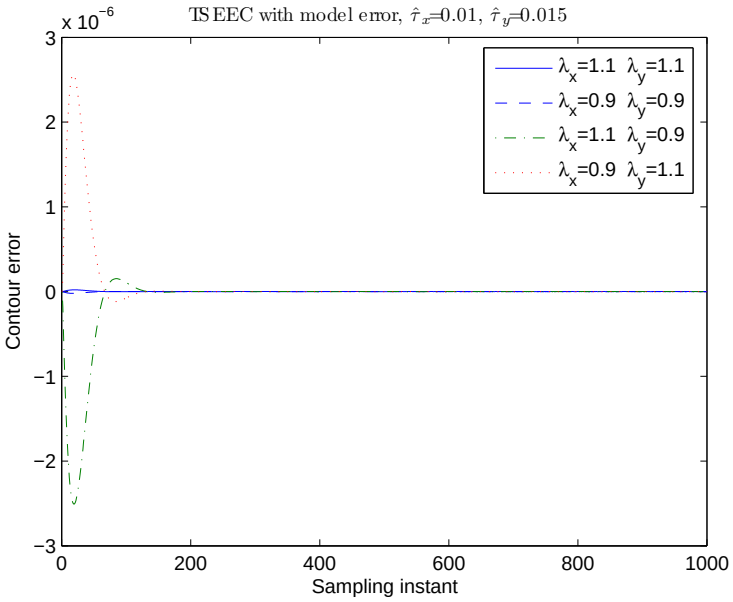

Figure 6.11: Linear contour error by TSEEC with ±10% estimation error in time constant.

The circular contouring performance when using ZPETC under these same four situations is shown in Fig. 6.12. The corresponding performance, under the same conditions, when using TSEEC is shown in Fig. 6.13. Here again, as can be clearly seen from the figures, the contour errors for TSEEC is almost 3 orders of magnitude smaller than that for ZPETC.

Although the Taylor series expansion error compensation approach and the feedforward controller both require a precise knowledge of the process, simulation results show that the contouring accuracy of the proposed TSEEC strategy is superior under conditions of modeling errors.

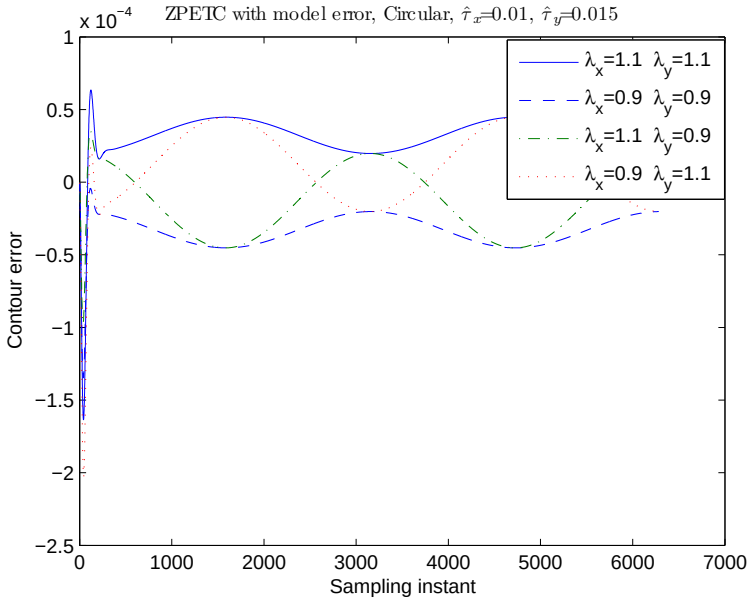

Figure 6.12: Circular contour error by ZPETC with ±10% estimation error in time constant.

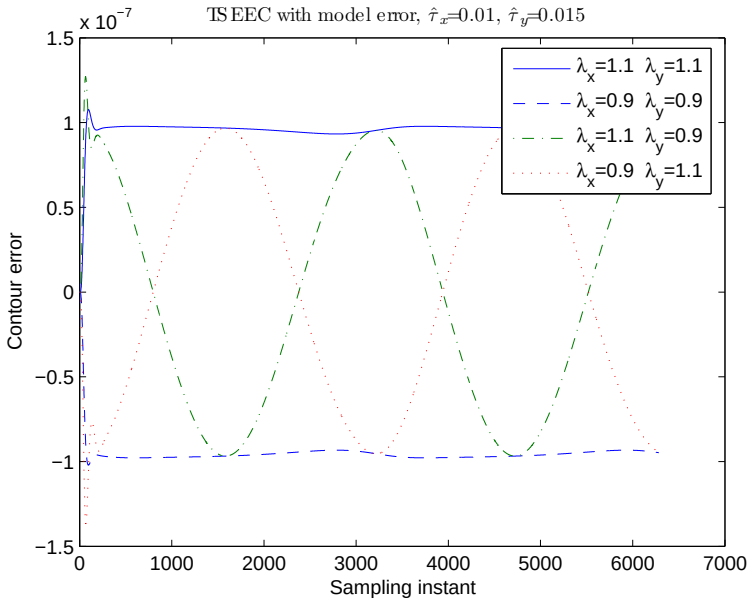

Figure 6.13: Circular contour error by TSEEC with ±10% estimation error in time constant.

Table 6.1: Coefficients for the X and Y model, for $d = 3$

$\frac{y(k)}{u(k)} = \frac{q^{-d}B(q^{-1})}{A(q^{-1})}$	q^0	q^{-1}	q^{-2}	q^{-3}
$B_x(q^{-1})$	0.00708911	0.00338940	0.00115112	0.0
$A_x(q^{-1})$	1.0	-1.79755510	1.05122750	-0.25367240
$B_y(q^{-1})$	0.00790339	0.00560520	0.00014294	0.0
$A_y(q^{-1})$	1.0	-1.52942945	0.64281390	-0.11338445

6.5 Experimental

Experiments were carried out to test the feasibility of the proposed TSEEC scheme. Because of imperfect knowledge of the dynamic models, and to cater to other effects such as external disturbances and measurement noise, some modifications are necessary.

6.5.1 Input-output model of servo drive

For the implementation of TSEEC on a mini-CNC, the dynamic models for the X and the Y axes are needed. As the open-loop servo drive contains an integrator, the standard open-loop system identification techniques cannot be easily used. The closed-loop identification method is therefore adopted to obtain the discrete time models for the X and the Y axes. The details of the closed-loop identification experiment are discussed in the Appendix. The model used for both the X and the Y axes is the following discrete-time model

$$\frac{y(k)}{u(k)} = \frac{q^{-d}B(q^{-1})}{A(q^{-1})} \tag{6.5.1}$$

The identified models for both the X and the Y axes are listed in Table 6.1.

6.5.2 Modifications to TSEEC for real implementation

Among several new issues encountered in the experiments, the first is the choice of the compensation goal for a discrete-time system with a time delay more than one sampling period. The second is the use of low-pass filters to filter out high-frequency components generated in the compensation signals. The third is the compensation gain which is used to adjust the amount of the compensation signal to be applied to the reference position inputs.

TSEEC is basically a model-based compensation method. In the simulation discussed in the previous sections, the time delay used was $d = 1$. The identified models, as listed in Table 6.1, however, have a time delay of $d = 3$. To cope with this increased time delay, it is necessary to modify the original compensation scheme.

During the initial trials of the experiments, the TSEEC compensation had the goal of making the contour error canceled out at the $(k + d)$ instant, or

$$\varepsilon(k + d) = 0 \tag{6.5.2}$$

However, the resulting compensation signals generated cause serious chattering. It was thus necessary to take measures to address the chattering problem. Two factors were found to have impact on the chattering. One is the predefined contour error dynamics and the other is the high-frequency components in the compensation signals.

With the time delay $d \geq 1$, the contour error can be expressed as

$$\varepsilon^*(k + d) = f(r_x^*(k), r_y^*(k)) \tag{6.5.3}$$

The task of TSEEC was re-adjusted to reduce the contour error such that the new compensation target is satisfied at the $(k+d)$ instant. More specifically, rather than attempting to make the contour error zero at the $(k+d)$ instant zero (Eq. 6.5.2), the new compensation target is to make the moving average of the contour errors in the next d steps (from $(k+1)$ to $(k+d)$ instant) zero. The contour error dynamics can be expressed as

$$P(q^{-1})\varepsilon(k+d) = (1 + p_1 q^{-1} + \ldots + p_{d-1} q^{d-1})\varepsilon(k+d) = 0 \tag{6.5.4}$$

The 2^{nd} order polynomial $P(q^{-1})$, which defines the contour error dynamics, can be designed from a continuous-time second-order polynomial,

$$s^2 + 2\zeta\omega_n s + \omega_n^2 = 0 \tag{6.5.5}$$

where ζ and ω_n are the damping ratio and the natural frequency of the desired contour error dynamics respectively and s is the Laplace transform operator. In this project, the designed contour error dynamics are chosen to have a damping ratio of 0.707 and a natural frequency of 520 rad/s. The polynomial $P(q^{-1})$ is then obtained by discretizing the continuous-time polynomial in Eq. (6.5.5).

We now discuss the modification of the calculation of the compensation signals with $d = 3$ from the case of $d = 1$. The target of TSEEC is now to make the contour errors in the next d steps satisfy the error dynamics defined by $P(q^{-1})$. After applying the compensation signals, the contour error at the $(k+d)$ instant will be

$$\begin{aligned}\varepsilon(k+d) &= f(r_x(k), r_y(k)) = f(r_x^*(k) + \Delta r_x(k), r_y^*(k) + \Delta r_y(k)) \\ &= \varepsilon^*(k+d) + A\Delta r_x(k) + B\Delta r_y(k) \end{aligned} \tag{6.5.6}$$

where $\varepsilon(k+d)$ and $\varepsilon^*(k+d)$ are the compensated and uncompensated contour errors respectively.

It is noted that the value of $\varepsilon(k+d)$ should be chosen such that the moving average of the contour errors, from $(k+1)$ to $(k+d)$ instant, is equal to zero, or

$$P(q^{-1})\varepsilon(k+d) = \varepsilon(k+d) + p_1\varepsilon(k+d-1) + \ldots + p_{d-1}\varepsilon(k+1) = 0 \tag{6.5.7}$$

Combining Eqs. (6.5.6) and (6.5.7), we have

$$\begin{aligned} & -(\varepsilon^*(k+d) + p_1\varepsilon(k+d-1) + \ldots + p_{d-1}\varepsilon(k+1)) \\ = \;& A\Delta r_x(k) + B\Delta r_y(k) \end{aligned} \tag{6.5.8}$$

Denote

$$\eta = \varepsilon^*(k+d) + p_1\varepsilon(k+d-1) + \ldots + p_{d-1}\varepsilon(k+1) \tag{6.5.9}$$

Simply substituting ε by η in Eq. (6.2.24), the compensation signals for the linear contour can be obtained from

$$\begin{pmatrix} b_{11}\sin\theta & -b_{21}\cos\theta \\ \frac{\cos\theta}{b_{21}} & \frac{\sin\theta}{b_{11}} \end{pmatrix} \begin{pmatrix} \Delta r_x(k) \\ \Delta r_y(k) \end{pmatrix} = \begin{pmatrix} -\eta \\ 0 \end{pmatrix} \tag{6.5.10}$$

Similarly, substituting ε by η in Eq. (6.2.34), the compensation signals for circular contours can be obtained from

$$\begin{pmatrix} A & B \\ -\frac{B}{b_{21}^2} & \frac{A}{b_{11}^2} \end{pmatrix} \begin{pmatrix} \Delta r_x(k) \\ \Delta r_y(k) \end{pmatrix} = \begin{pmatrix} -\eta \\ 0 \end{pmatrix} \tag{6.5.11}$$

6.5.3 Low-pass filter

As the identified models are imperfect, and also because of the existence of external disturbances and measurement noise, it was found that the compensation signals as obtained using Eq. (6.5.10) and (6.5.11) contain undesirable high-frequency components. This situation is not encountered in the noise-free environment of simulation. As the compensation signals, $\Delta r_x(k)$ and $\Delta r_y(k)$, are directly superimposed onto the reference position inputs of the axial servo loops, the high-frequency components resulted in serious chattering. This not only is harmful to the mechanical transmission system of the machine, but also negates the efforts of reducing contour errors.

It was therefore found necessary to pass these compensation signals through a low-pass filter. A low-pass filter, on one hand, can remove the high-frequency components, but on the other, can also introduce a phase lag, which is harmful to the stability of the closed-loop system. The cutoff frequency of the low-pass filter cannot be chosen to be too low. In the project, we used a second-order low-pass filter with a cutoff frequency set at 30 Hz. The filter was designed by the Butterworth method in the MATLAB toolbox.

6.5.4 Compensation gain

To address the problem of imperfectness of the dynamic models and external disturbances, a compensation gain is used to adjust the strength of the compensation signals that are actually applied to the real system. The actual reference position input, after compensation, is then given by

$$r(k) = r^*(k) + \alpha \Delta r(k) \tag{6.5.12}$$

where α is the compensation gain. $\alpha = 0$ represents the case when no compensation is applied and $\alpha = 1$ the case for perfect compensation when perfect knowledge of the dynamics is available and there are no other unknown disturbances. The best value of α, for a particular situation, as will be shown later, can be found experimentally.

6.5.5 Experimental results

The two proportional controllers were set to $K_{px} = 10$ and $K_{py} = 8$ to make the two axial dynamics unmatched. Without any compensation, a linear contour at 45° to the X-axis and a circular contour of radius 40 mm were generated with the same feedrate of 40 mm/s. The results are shown in Fig. 6.14. It is noticed that in the steady state, the contour error for the linear contour is about 15 μm as shown in Fig. 6.14 (a) and the maximum for the circular contour is 15 μm as shown in Fig. 6.14 (b).

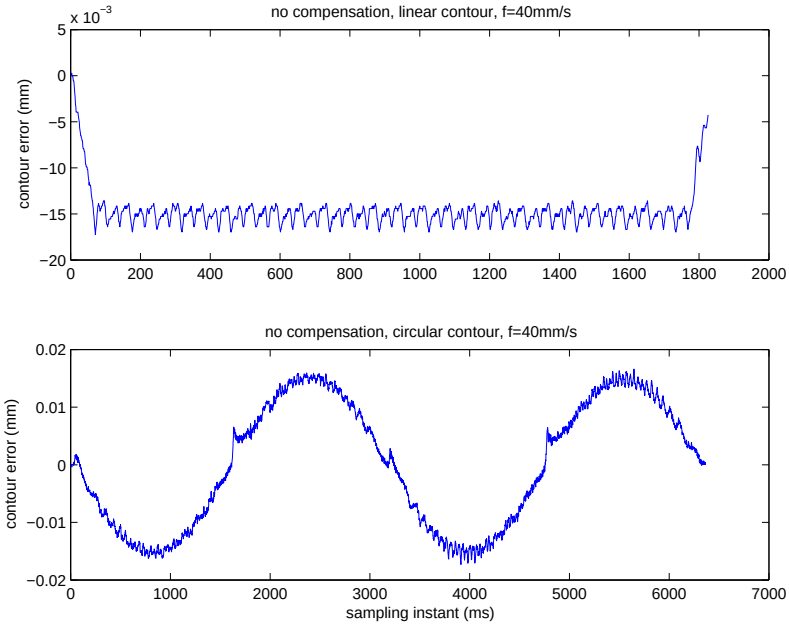

Figure 6.14: Experimental circular contour error without compensation, $f = 40$ mm/s: (a) Linear contour and (b) Circular contour.

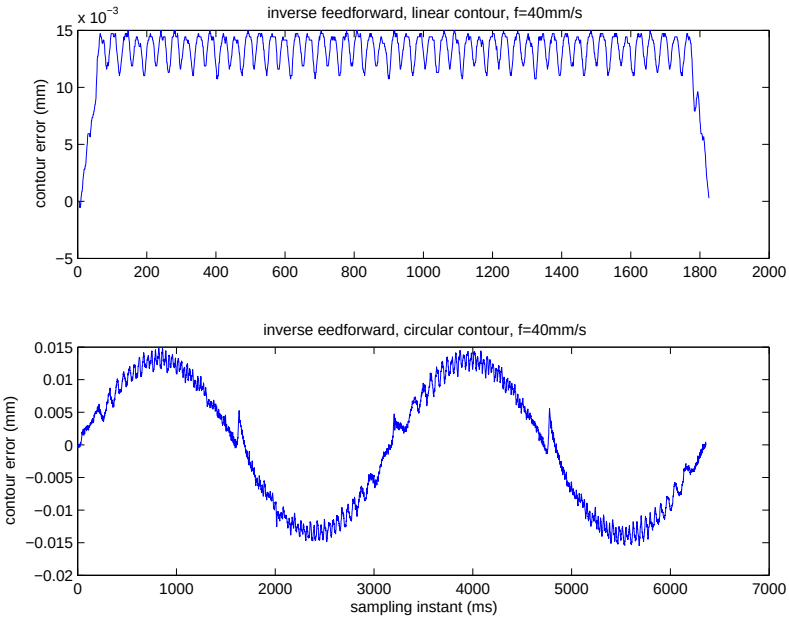

Figure 6.15: Experimental circular contour error by inverse feedforward compensation, $f = 40$ mm/s: (a) Linear contour and (b) Circular contour.

Performance of inverse feedforward compensation

It is noted from Table 6.1 that the identified models for both the X and the Y axes are of the minimum phase type. In this instance, therefore, a ZPETC feedforward compensator which is meant for non-minimum phase plants is not suitable or necessary. A simpler inverse feedforward compensation, with a transfer function the inverse of the closed-loop servo drive, is thus used instead. With this inverse feedforward compensation, the contour errors for the same linear and circular contours were generated and the results are shown in Fig. 6.15. It can be noted from the figure that for both the linear and circular contours, the orientation, or phase, of the contour errors are just opposite to that of Fig. 6.14 when there is no compensation. The magnitude of the steady-state linear contour errors have been reduced to only marginally from 15 μm to 14 μm and the maximum circular contour errors also by the same amount to 14 μm. The contouring accuracy is thus only slightly improved by using feedforward compensation which is designed to be the inverse of the closed-loop transfer function. Although theoretically the output should follow the reference input, modeling error and external disturbances made the feedforward compensation ineffective in practice.

Performance of TSEEC

The proposed TSEEC compensation was then tested for its contouring compensation error performance. The moving average contour errors, defined by Eq. (6.5.7), was used as the compensation target to cope with the increased time delay. Low-pass filters were used to filter out the high-frequency components in the compensation signals, $\Delta r_x(k)$ and $\Delta r_y(k)$.

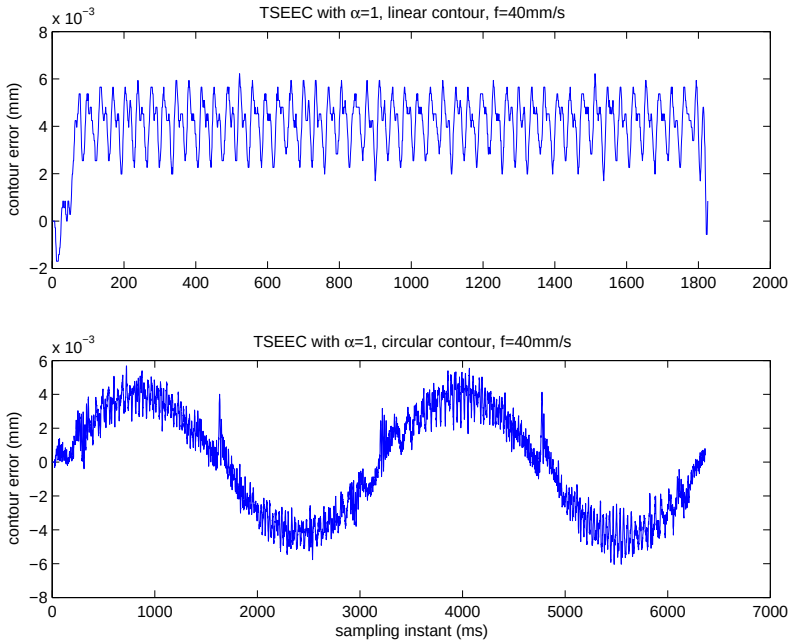

Figure 6.16: Experimental circular contour error by TSEEC with $\alpha = 1$ and $f = 40$ mm/s: (a) Linear contour and (b) Circular contour.

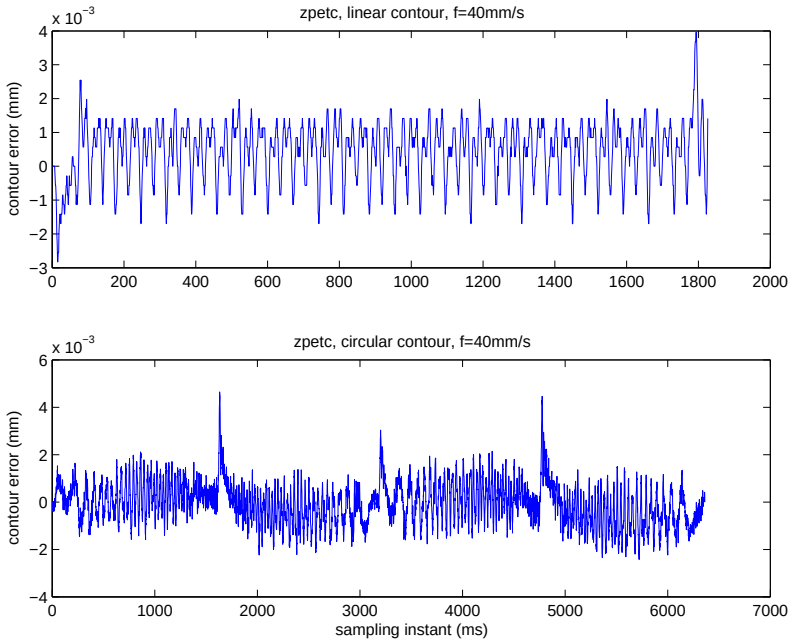

Figure 6.17: Experimental circular contour error by TSEEC with $\alpha = 0.6$ and $f = 40$ mm/s: (a) Linear contour and (b) Circular contour.

Compensation gains, as described by Eq. (6.5.12), were also used to cope with modeling error. When TSEEC was applied with $\alpha = 1$, the resulting contour errors are as shown in Fig. 6.16 for both the linear and circular contours. When these are compared with those in Fig. 6.14, the case when $\alpha = 0$ with no compensation, it can be noted that both the steady-state linear contour error and the maximum circular contour error have been reduced from about 15 to 4 μm. While the magnitudes have been reduced, there are still contour errors present which have opposite orientations, or phases, as compared with the case of $\alpha = 0$.

This indicates that, due to modeling errors, overcompensation has resulted. It is thus reasonable to deduce that there exists a compensation gain between these two values of α for which the contour error will be optimum. By trial and error, this value was found as $\alpha = 0.6$. The corresponding contour errors for the linear contour are shown in Fig. 6.17 (a). The contour error has been reduced to around ± 2 μm. Considering that the resolution of the position feedback encoders used was 0.4 μm, these are very good results.

For the circular contour, except for the error spikes at quadrant positions caused by stiction at the reversal of velocities or moving from standstill, the magnitude of the contour errors are now also within a band of ± 2 μm, largely reduced from ± 15 μm in the case of no compensation. This clearly illustrates the effectiveness of the proposed TSEEC scheme in improving the contouring accuracy for both linear and circular contours.

The corresponding compensation signals are shown in Figs. 6.18 and 6.19 for the linear and the circular contours respectively. It can be clearly seen in part (a) and

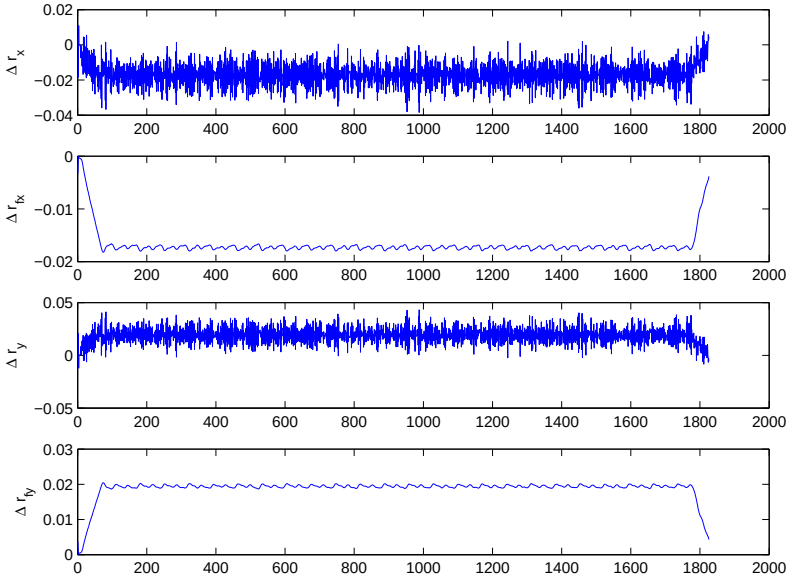

Figure 6.18: Linear contour compensation signals by TSEEC with $\alpha = 0.6$ and $f = 40$ mm/s: (a) Δr_x, (b) Δr_{fx}, (c) Δr_y and (d) Δr_{fy}

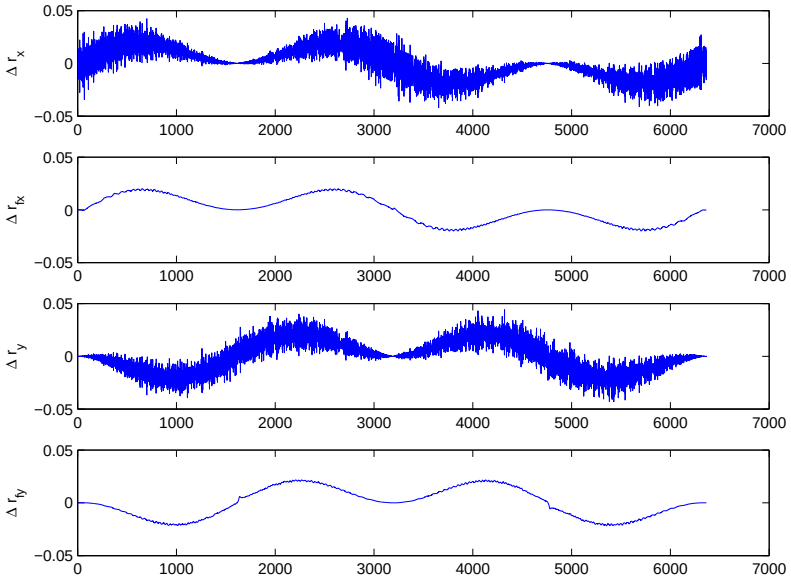

Figure 6.19: Circular contour compensation signals by TSEEC with $\alpha = 0.6$ and $f = 40$ mm/s: (a) Δr_x, (b) Δr_{fx}, (c) Δr_y and (d) Δr_{fy}

(c) of these figures that the original compensation signals contain high-frequency components, which cause undesirable chattering. To avoid chattering, these signals were passed through low-filter filters. The filtered compensation signals, which were actually applied to the reference position inputs, are shown in parts (b) and (d) of Figs. 6.18 and 6.19. It can be noted that these filtered compensation signals are now very smooth.

For TSEEC compensation for circular contours, we proposed two compensation schemes, scheme 1 and scheme 2. The superiority of scheme 1 over scheme 2 is held only when the dynamical model is perfect. In the actual implementation, however, due to the modeling error and external disturbances, scheme 1 does not show superiority in contouring performance over scheme 2 and there is no noticeable differences between the two schemes. We therefore do not differentiate between the two schemes in the experiment.

6.6 Conclusion

In this chapter, a new compensation strategy (TSEEC), which is based on the Taylor series expansion, is used to improve contouring accuracy in CNC machines. In TSEEC, the contour error compensation problem is formulated as a Taylor series expansion problem, in which the value of the contour error is expanded around the reference position points along the path of motion and the compensation components are calculated as the deviation from the reference points.

Simulation results show that, with perfect knowledge of axial dynamics, the proposed strategy can completely cancel out the contour errors for both linear and circular

contours even with mismatches in axes' dynamics. Simulation results also show the robustness of the proposed TSEEC approach under conditions of imperfect knowledge of axial dynamics. Even when the modeling error is about 10%, the approach can still maintain good performance, with the contour errors reduced by four orders of magnitude as compared to an uncompensated system.

A comparison made between this approach and the zero phase error tracking controller (ZPETC) also shows that this proposed TSEEC approach performs significantly better in the presence of modeling errors.

The proposed (TSEEC) compensation approach has been tested experimentally on a real mini-CNC machine. Due to modeling errors, external disturbances, and measurement noise, some measurements and adjustments need be made before the implementation of the TSEEC scheme. The first is the choice of the compensation goal for a discrete-time system with a time delay more than one sampling period. Rather than to attempt to make the contour error at the $(k+d)$ instant zero, where k is the current sampling instant and d is the time delay in a discrete-time system, the goal is adjusted to make the moving average of the contour errors in the future d steps zero. The second is the use of a low-pass filter to filter out high-frequency components generated in the compensation signals. The third parameter determined experimentally by trial and error is the value of the compensation gain used to adjust the amount of the compensation signal to be applied to take care of modeling errors and other disturbances. It was shown experimentally that a proper choice of this compensation gain can result in much improved contouring performance, with the contour errors reduced to within a few times the basic length unit(BLU).

Chapter 7

Improving contouring accuracy by an integral sliding mode controller

In this chapter, an integral sliding mode controller (ISMC), based on input-output models, is proposed as a refinement from a two-degree-of-freedom controller with independent objectives for tracking and regulation. Thus, the knowledge of pole placement can be utilized in ISMC. The robustness was improved by a disturbance estimation, which results in an equivalent control. To eliminate the problem of chattering, two measures adopted are the appropriate choice of the sliding surface and the integral switching function. It was found that the choice of a slower natural frequency of the sliding surface dynamics than that of the open loop can ease the problem of chattering. The choice of the integral gain k_i is based on the two-degree-of-freedom controller. The root locus is used in assisting in the choice of an appropriate value for k_i to ensure the closed-loop stability. The proposed ISMC was implemented and experimentally tested in a CNC machine. The contouring accuracy of the CNC machine was found to be greatly improved by the proposed ISMC. Furthermore, no chattering was observed, which is beneficial to the machine actuators.

7.1 Introduction

To improve the contouring accuracy for multi-axis CNC machines, the straightforward approach is to improve the tracking accuracy for each individual axis, thus the overall contouring accuracy can be ensured. The zero phase error tracking controller(ZPETC) proposed by Tomizuka [56] is based on this idea. ZPETC is basically a feedforward controller. For minimum phase plants, the feedforward controller can be designed as the inverse of the plant. The transfer function from the reference input to the output then becomes unity. With a perfect model, the tracking error can be zero, thus resulting in zero contour errors. For non-minimum phase plants, ZPETC is designed to be an approximated inverse of the plant. The transfer function from the reference input to the output is approximately unity at low frequencies. When the model is perfect, the output should closely follow the reference input, which results in very small tracking errors, and indirectly results in very small contour errors. The performance of inverse or ZPETC controllers, however, heavily depends on the quality of the model. With imperfect models, the improvement in contouring accuracy with inverse or ZPETC control is negligible [30].

To overcome model uncertainties and external disturbances which are inevitable in practice, one promising method to reduce tracking errors is sliding mode control (SMC). Sliding mode control is well known for its robustness against model uncertainties and external disturbances [60]. In continuous-time systems, the robustness can be ensured by employing a switching function with infinite frequency. This switching function can drive the system onto the sliding surface, and remain on it thereafter.

As servo controllers in CNC machines are implemented by digital computers, it is

necessary to investigate the discrete-time sliding mode controller (DSMC). Limited by the finite sampling interval, the switching cannot occur at infinite frequency as in the case of continuous-time SMC. The switching can only happen at the sampling instant. This means that when the system dynamics cross the sliding surface between the sampling instants, the control signals cannot immediately take actions to make the system remain on the sliding surface. Therefore, in discrete-time sliding mode control, the system will, at best, stay within the neighborhood of the sliding surface, rather than just remain on the sliding surface [19, 26].

One design of SMC consists of an equivalent control and a switching function. The equivalent control is solved from the sliding surface $s_k = 0$ [16], while the switching function is designed to drive the system onto the sliding surface when it deviates from the surface. Sliding mode control is sometimes called variable structure control (VSC) because the switching function is usually a sign function or a saturation function. Depending on which side of the sliding surface the system states lie in, SMC (or VSC) controllers will have different structures .

To maintain the robustness of the SMC, the estimation of the disturbance is a necessity in the design of SMC. Eun *et al.* [18] developed an approach which combines the discrete variable structure control (DVSC) and the decoupled disturbance compensator(DDC) to control a single axis for a CNC machine. Later Kim *et al.* [26] improved it by modifying the sliding surface. Sun *et al.* [52] designed a variable structure repetitive control based on a discrete-time input-output model with a time delay and had the current sliding function value estimate the sliding function at the d^{th} step in the future.

In the aforementioned papers, the SMC employed discontinuous elements in the control law. Robustness is ensured at the price of introducing high-frequency components in the control signals, which causes "chattering" and is undesirable for actuators. For practical reasons, it is desirable to keep the system dynamics remaining on the sliding surface without introducing discontinuous elements into control signals. Integral sliding mode control [9, 7] is intended to address the problem of chattering.

There are reported applications of SMC to improve the contouring accuracy for CNC machines. Altintas *et al.* [2] presented an adaptive sliding mode control for the control of high speed feed drives. Chen *et al.* [7] proposed two integral sliding mode controllers based on different characteristics of the model dynamics. Comparing with conventional SMC, actuator chattering was eliminated by introducing an integral action.

In this chapter, an integral sliding mode controller (ISMC), based on input-output models, is proposed as a refinement from a two-degree-of-freedom controller with independent objectives for tracking and regulation. Thus, the knowledge of pole placement can be utilized in ISMC. The robustness was improved by a disturbance estimation, which results in the equivalent control. To eliminate the problem of chattering, two measures adopted are an appropriate choice of the sliding surface and the use of an integral switching function. It was found that the choice of a natural frequency of the sliding surface dynamics that is lower than that of the open-loop transfer function can ease the problem of chattering. The choice of k_i, the integral gain in ISMC, is based on the two-degree-of-freedom controller. The root locus is used to assist in the choice of an appropriate value for k_i to ensure closed-loop stability.

The proposed ISMC was implemented and experimentally tested in a CNC machine to evaluate its performance in improving contouring accuracy. The contouring accuracy of the CNC machine was found to be greatly improved by the proposed ISMC. Furthermore, no chattering was observed, which is beneficial to the machine actuators.

The rest of this chapter is organized as follows: Section 7.2 briefly introduces the identified axial open-loop model used in this chapter. Section 7.3 discusses the design of a two-degree-of-freedom controller based on the input-output model and Section 7.4 introduces the discrete-time sliding mode control and derives the equivalent control based on the estimation of disturbance. The choice of the sliding surface is discussed in Section 7.5 followed by a presentation of the integral sliding mode control in Section 7.6. Next, the experimental results are discussed in Section 7.7 and followed by the conclusion in Section 7.8.

7.2 Dynamic model

The dynamics of an axial drive system can be expressed as a discrete-time input-output model

$$A(q^{-1})y_k = q^{-d}B(q^{-1}) + w_k \tag{7.2.1}$$

where d is the time delay of the discrete-time system, and w_k a lumped disturbance term which includes external disturbances and model uncertainties. The numerator $B(q^{-1})$ and the denominator $A(q^{-1})$ are the time-shift polynomials of order m and n respectively,

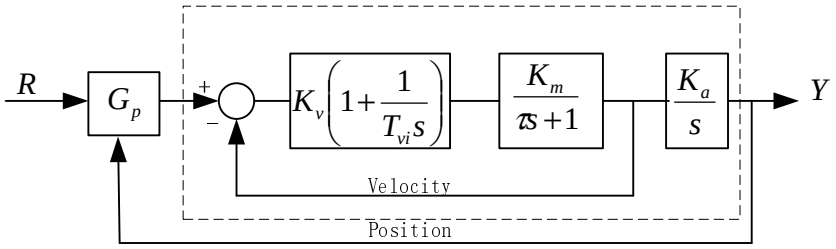

Figure 7.1: Block diagram for axial servo drive system.

$$B(q^{-1}) = b_0 + b_1 q^{-1} + b_2 q^{-q} + \cdots + b_m q^{-m} \tag{7.2.2}$$

$$A(q^{-1}) = 1 + a_1 q^{-1} + a_2 q^{-2} + \cdots + a_n q^{-n} \tag{7.2.3}$$

The open-loop dynamic models of the X and the Y axes of the mini-CNC, as enclosed in the dashed line in Fig. 7.1, were obtained by the closed-loop identification method [31]. The details of the closed-loop identification experiment are discussed in the Appendix. The identified models for the X and the Y axes are listed in Table 6.1.

7.3 Two-degree-of-freedom (RST) controller

Consider the two-degree-of-freedom controller as shown in Fig. 7.2. Given a certain open-loop system model, the two degrees of controller design basically refer to how the choice of the regulation and the tracking performance of the closed-loop system is made. R, S and T in Fig. 7.2 are the time shift polynomials of the designed controller. For a specific model, different design methods will result in different sets of R, S and T [31]. For easy reference, the two-degree-of-freedom controller will be called an RST controller.

In this project, the sliding mode controller will be designed step by step from a

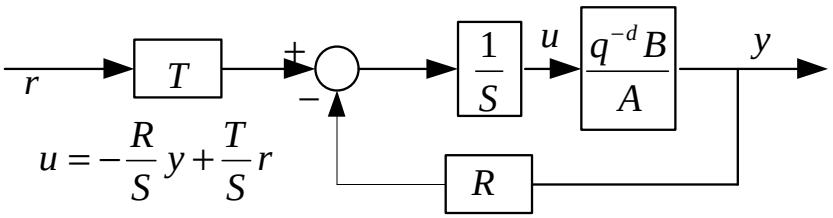

Figure 7.2: Two-degree-of-freedom controller (RST)

standard two-degree-of-freedom (RST) controller followed by an equivalent controller and finally an integral sliding mode controller. At each step, a controller is designed and tested and the subsequent controller designed based on the previous one. That is to say, the equivalent controller is based on the RST controller, and the integral sliding mode controller based on the equivalent controller. Rather than designing a new sliding mode controller to be tested on a real machine in just one step, the use of intermediate steps is helpful in trouble shooting.

As the identified models of the servo drive dynamics are of the minimum phase type, the models can be use to designed an RST controller with independent objectives of tracking and regulation. The control law is given by

$$u_k = -\frac{R}{S}y_k + \frac{T}{S}r_k \tag{7.3.1}$$

which contains a feedback component $-\frac{R}{S}y_k$ and a feedforward component $\frac{T}{S}r_k$.

With a choice of a $T = P$, where P is a polynomial of the sliding surface to be described later, the choice of R and S can be determined by pole placement controller design. These polynomials are designed such that

$$A(q^{-1})S(q^{-1}) + q^{-d}B(q^{-1})R(q^{-1}) = B(q^{-1})P(q^{-1}) \tag{7.3.2}$$

It is noted that for such an RST design with independent objectives of tracking and regulation, the closed-loop poles contain the factor $B(q^{-1})$. This means that the open-loop zeros are canceled out by the RST controller.

Assuming that lumped disturbance $w_k = 0$ and with an accurate model, the control law given by Eq. (7.3.1) together with the condition of Eq. (7.3.2) will result in an overall closed-loop system transfer function described by q^{-d}. This means that if there are no disturbances and plant/model mismatch, the system output y_k should follow the reference input r_k with a time delay of d steps. However, due to the imperfectness in the identified models and the presence of external disturbances, perfect following cannot be achieved as will be illustrated later in the experiments.

7.4 Sliding mode controller design

7.4.1 Derivation of SMC

Sliding mode control (SMC) is known for its robustness to disturbances and model uncertainties. SMC is sometimes called variable structure control (VSC) as it is quite common to use a variable structure to drive the system dynamics to achieve the sliding mode [26]. The control signal consists of equivalent control and sliding mode control [50].

$$u_k = u_k^{eq} + u_k^{sw} \tag{7.4.1}$$

The equivalent control component u_k^{eq} relies on the dynamics of the system and the sliding surface s_{k+d}. This can be determined by solving [26, 6]

$$s_{k+d} = 0. \tag{7.4.2}$$

The component u_k^{sw} is the switching function that drives the system onto the sliding surface.

For continuous-time SMC, the sliding mode can be achieved by assuming that infinitely fast switching between two different control structures is possible.

In discrete-time systems, however, switching of the variable control structures can only happen at discrete sampling instants. If the system dynamics cross the sliding surface in-between these sampling, the controller cannot take immediate measures to drive the system back to the sliding surface. Thus, the system dynamics cannot be kept exactly on the sliding surface at all time. Rather, the switching function will result in a zigzag route around the defined sliding surface.

For a discrete-time system defined as Eq. (7.2.1), the sliding surface is defined as

$$s_{k+d} = P(q^{-1})(r_{k+d} - y_{k+d}) \tag{7.4.3}$$

where $P(q^{-1})$ is a polynomial that defines the sliding surface. The choice of zeros of $P(q^{-1})$ is found to have great influence on chattering, and this will be discussed in later sections.

The equivalent control can be obtained by finding a control that can make the sliding function equal to zero. The polynomial $P(q^{-1})$ is written as

$$P(q^{-1}) = A(q^{-1})E(q^{-1}) + q^{-d}F(q^{-1}) \tag{7.4.4}$$

As $P(q^{-1})$ is provided by the designer and $A(q^{-1})$ is known from the model, $E(q^{-1})$ and $F(q^{-1})$ can be solved by the Bezout polynomial equation [31].

Substituting Eqs. (7.2.1) and (7.4.4) into Eq. (7.4.3), we have

$$\begin{aligned} s_{k+d} &= Pr_{k+d} - Py_{k+d} \\ &= Pr_{k+d} - EBu_k - Ew_{k+d} - Fy_k \end{aligned} \tag{7.4.5}$$

Assume the axial drive is a minimum phase system. For the X and Y axis identified in this project, both models are of the minimum phase type.

Therefore, the equivalent control can be derived as

$$u_k^{eq} = (EB)^{-1}(Pr_{k+d} - Fy_k - Ew_{k+d}) \tag{7.4.6}$$

Note from Eq. (7.4.6) that this equivalent control is not realizable as the calculation of the control at the current step k involves knowledge of the disturbance in the next (or future) d instants. To circumvent this difficulty, an estimate of the disturbance in the future d steps by making the following assumption

$$Ew_{k+d} = Ew_k \tag{7.4.7}$$

This assumes that the moving average of the disturbances in the future d instants is

the same as that of the disturbances in the immediate past d instants including the current instant. This assumption should be reasonable as the sampling interval is small in comparison with the dynamics of the servo control systems (e.g. 1 ms in this project). The lumped disturbance w_k usually will also not change so abruptly within d sampling intervals. The disturbance at the instant k can then be estimated as

$$w_k = A(q^{-1})y_k - B(q^{-1})u_{k-d} \tag{7.4.8}$$

Eq. (7.4.8) implies that this estimated disturbance is actually the prediction error for the current instant. Influenced by the external disturbances and measurement noise, the estimate of disturbances can contain high-frequency components. When applied directly in the equivalent control law, the estimated disturbance is then likely to result in chattering. A feasible way to address this problem of chattering is to pass these estimates through a low-pass filter as suggested by Beghi *et al.* [6].

With the assumption made on the disturbances and the use of a low-pass filter, the realizable controller is obtained as

$$u_k^{eq} = (EB)^{-1}(Pr_{k+d} - Fy_k - Ew_{fk}) \tag{7.4.9}$$

where $w_{fk} = Q(q^{-1})w_k$ is the estimated disturbance filtered by the low-pass filter $Q(q^{-1})$.

7.4.2 Relationship between RST controller and equivalent control action

Compared with the RST controller in Eq. (7.3.1), the equivalent control action u_k^{eq} of Eq. (7.4.9) contains the estimate of disturbances, where $S = EB$, $R = F$ and $T = P$. Without disturbance w_k, the equivalent control will be exactly the same as the RST controller with independent objectives for tracking and regulation. With the estimate of disturbances, the actual closed-loop dynamics will be closer to the dynamics that are defined by the predefined polynomial $P(q^{-1})$. In this situation, an inappropriately chosen set of zeros for $P(q^{-1})$ could cause serious chattering depending on the actual servo drive dynamics. If the closed-loop natural frequency ω_{close} is chosen to be close to that of the open-loop ω_{open}, serious chattering will result, which is harmful to the actuators.

7.5 Choice of the sliding surface

The sliding surface is defined by the polynomial $P(q^{-1})$ which is found to have a great impact on the performance of sliding mode control. To the best knowledge of the author, there has been no discussion on the relationship between the chattering problem and the choice for $P(q^{-1})$. It was found in this project that a set of carefully chosen zeros of $P(q^{-1})$ can eliminate the chattering problem, which is the main disadvantage of sliding mode control.

$P(q^{-1})$ defines part of the closed-loop poles when the system is closed by the SMC controller. The locations of the closed-loop poles determine the damping ratio and natural frequency of the closed loop. In the left hand side of the s plane, those

poles closest to the imaginary axis are the dominant poles. It was found from the experiments that the dominant natural frequency of the closed-loop system should be chosen to be slower than that of the open loop in order to avoid undesirable chattering. From the identified models of the X and Y axes, the damping ratio and natural frequency of the open loop can be calculated. For a open-loop system with multiple poles, it is only necessary to check its dominant poles, as the responses introduced by auxiliary poles decay faster than those introduced by dominant poles.

It is more convenient to analyze the dynamics in the s (Laplace transform operator) domain. As the identified models are of discrete-time forms, it is necessary to convert them into their continuous-time counterparts. Suppose a discrete-time second-order polynomial is discretized from its continuous-time counterpart and the sampling interval is T_s. The zeros in the s plane can be mapped into the zeros in the z (z-transform operator) plane by

$$e^{sT_s} = z \tag{7.5.1}$$

The the continuous-time second-order polynomial

$$s^2 + 2\zeta\omega_n + \omega_n^2 = 0, \qquad \zeta < 1 \tag{7.5.2}$$

has the zeros

$$s_{1,2} = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2} \tag{7.5.3}$$

Therefore, the corresponding zeros in the z plane can be determined as

$$\begin{aligned} z_{1,2} &= e^{-\zeta\omega_n T_s \pm j\sqrt{1-\zeta^2}\omega_n T_s} = e^{-\zeta\omega_n T_s} e^{\pm j\sqrt{1-\zeta^2}\omega_n T_s} \\ &= A\angle\phi \end{aligned} \tag{7.5.4}$$

Table 7.1: Determination of $P(q^{-1})$ from ζ_{close} and ω_{close}

axis	$A_1(q^{-1})$	ζ_{open}	ω_{open} (rad/s)	ζ_{close}	ω_{close} (rad/s)
X	$1 - 0.7976q^{-1} + 0.2537q^{-2}$	0.7221	949.8	0.707	520
Y	$1 - 0.5294q^{-1} + 0.1134q^{-2}$	0.8529	1276.2	0.707	520

where A is the magnitude, and ϕ is the phase angle of the zeros. Note that

$$(-\zeta\omega_n T_s)^2 + (\sqrt{1-\zeta^2}\omega_n T_s)^2 = (\ln A)^2 + \phi^2 = \omega_n^2 T_s^2 \tag{7.5.5}$$

we can then compute the damping ratio and natural frequency from

$$\omega_n = \frac{\sqrt{(\ln A)^2 + \phi^2}}{T_s} \tag{7.5.6}$$

$$\zeta = -\frac{\ln A}{\omega_n T_s} \tag{7.5.7}$$

In this project, the servo drive consists of an inner velocity feedback loop and and an outer position feedback loop. The transfer function for the identified velocity loop is denoted as $q^{-d}B(q^{-1})/A_1 q^{-1}$. The denominator $A_1(q^{-1})$ is used to calculate the open-loop damping ratio ζ_{open} and the natural frequency ω_{open}. The details for the X and the Y axes are listed in Table 7.1. The polynomial $P(q^{-1})$ is chosen such that the damping ratio ζ_{close} and the natural frequency ω_{close} for the closed loop are 0.707 and 520 rad/s, respectively. It is clear that the response of the closed loop is slower than that of the open loop.

After the closed-loop damping ratio and the natural frequency are chosen, the continuous-time system represented by

$$P(s) = s^2 + 2\zeta\omega_n + \omega_n^2 \tag{7.5.8}$$

is converted to its discrete-time equivalent [31, pp.50] as

$$P(q^{-1}) = 1 + a_1 q^{-1} + a_2 q^{-2} \tag{7.5.9}$$

where $a_1 = -2e^{-\zeta\omega_n T_s}\cos\left(\omega_n T_s\sqrt{1-\zeta^2}\right)$ and $a_2 = e^{-2\zeta\omega_n T_s}$. Discretizing Eq. (7.5.8), we obtain

$$P(q^{-1}) = 1 - 1.2921q^{-1} + 0.4794q^{-2} \tag{7.5.10}$$

This polynomial defines the sliding surface and is used for both the X and the Y axes.

7.6 Integral sliding mode control

7.6.1 Integral action

We have derived the equivalent control u_k^{eq} from the expression of $s_k = 0$. u_k^{eq} only defines the dynamics when the system is already on the sliding surface. When the system is off the sliding surface, a switching function is needed to drive the system onto the sliding surface. Various switching functions can be adopted for this purpose, the sign and saturation functions beding the common choices. However, although these methods can drive the system onto the sliding surface, chattering is likely to happen due to the discontinuity of the switching function and, consequently, the presence of high-frequency components in the control signal. Chattering is quite undesirable as it will accelerate the wear and tear of actuators. Many papers devoted to the application of SMC to real systems use integral sliding mode control (ISMC)

(or sliding mode control with integral action). Li and Wikander [32] proposed a discrete-time sliding mode controller with integral action to control linear motors for precision servo systems. Beghi *et al.* [6] designed an integral SMC based on an input-output model for a motorcycle engine. The analysis of stability with the change of the integral gain, k_i, was carried out with the help of the root locus. The main advantage of integral SMC is that it can drive the system to the sliding surface and stay on the surface thereafter while avoiding the undesirable chattering problem.

The proposed integral SMC scheme used in the work described here is shown in Fig. 7.3. Part (a) is the original block diagram, and part (b) is a simplification of part (a). The switching function is simply chosen as the integral of s_k and given by.

$$u_k^{sw} = \frac{k_i}{1 - q^{-1}} s_k \tag{7.6.1}$$

7.6.2 Choice of integral coefficient k_i

With the addition of integral action, the closed-loop poles are now changed. The choice of k_i has to be carefully made to ensure closed-loop system stability. However, there is no guarantee that a stable solution will always exist. The root locus technique can be used to assess the feasibility and suitability of possible values of k_i.

From Fig. 7.3, assuming there is no disturbance, i.e. with $w_k = 0$, the transfer function from r_{k+d} to y_k can be further simplified to

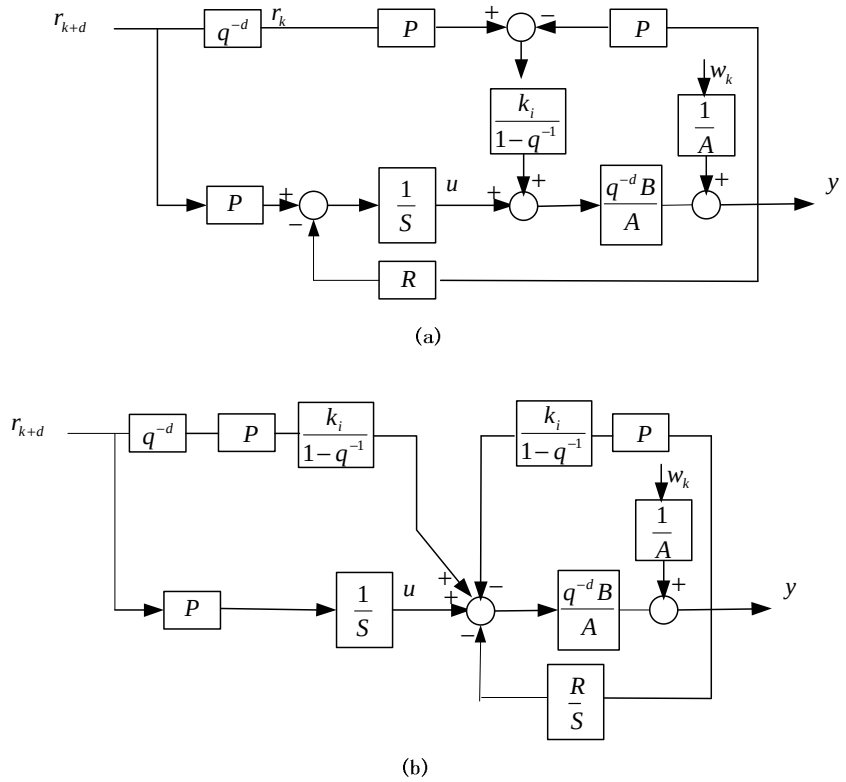

Figure 7.3: Stability of integral sliding mode control: (a) original, (b) simplified

$$\begin{aligned}\frac{y(k)}{r(k+d)} &= \frac{q^{-d}k_iPS + P(1-q^{-1})}{S(1-q^{-1})}\frac{q^{-d}BS(1-q^{-1})}{(AS+q^{-d}BR)(1-q^{-1})+q^{-d}k_iBPS} \\ &= \frac{q^{-d}k_iPS + P(1-q^{-1})}{S(1-q^{-1})}\frac{q^{-d}BS(1-q^{-1})}{(1-q^{-1})BP+q^{-d}k_iBPS} \\ &= \frac{q^{-d}k_iPS + P(1-q^{-1})}{1}\frac{q^{-d}}{(1-q^{-1})P+q^{-d}k_iPS}\end{aligned} \tag{7.6.2}$$

The closed-loop poles are determined by $BP((1-q^{-1})BP+q^{-d}k_iS)$. As BP is known, the additional poles are introduced by $(1-q^{-1})+q^{-d}k_iS$. Further simplification can be carried out if

$$P_{ki} = q^{-d}k_iS + (1-q^{-1}) \tag{7.6.3}$$

is stable. That means that once the RST controller is designed, k_i must be chosen such that P_{ki} is stable. If P_{ki} is stable, then the transfer function from r_{k+d} to y_k is q^{-d}. This means that the output follows the reference input with a time delay of d sampling instances.

As an illustration, we use the ISMC controller of the X axis to demonstrate how the root locus is used in assisting in the choice of an appropriate value of k_i. Referring to Eq. (7.6.3), S is a fourth-order polynomial as given in Table 7.2. Substituting S into P_{ki}, we obtain

$$P_{ki} = 1 - q^{-1} + q^{-d}k_i(s_0 + s_1q^{-1} + s_2q^{-2} + s_3q^{-3} + s_4q^{-4}) \tag{7.6.4}$$

Converting P_{ki} into a polynomial P_2 for the analysis of root locus, we have

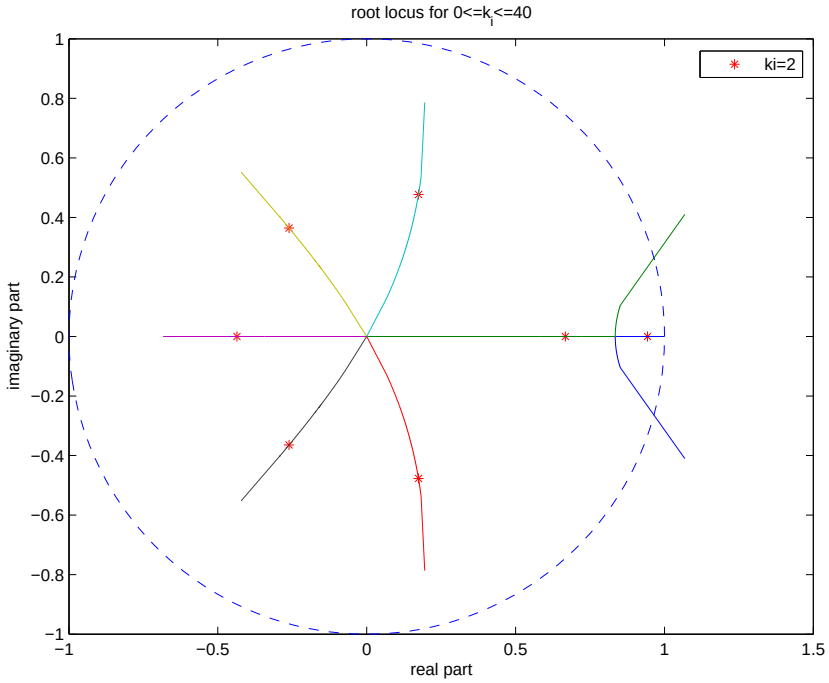

Figure 7.4: Root locus for $0 \leq k_i \leq 40$ and poles when $k_i = 2$ for the X axis ISMC

$$P_2 = 1 + \frac{k_i(s_0 q^4 + s_1 q^3 + s_2 q^2 + s_3 q + s_4)}{q^{d+3}(q-1)} \tag{7.6.5}$$

where the time delay $d = 3$ in this project.

The root locus, drawn using MATLAB with the value of k_i varying between 0 and 50 is shown Fig. 7.4. It can be seen that there are seven branches of the root loci, with six of them starting from the origin and the 7^{th} from $(1, 0)$. When $k_i = 2$, the closed-loop poles introduced by Eq. (7.6.3) are shown as "*" in Fig. 7.4. At this value of k_i for the X axis, it is clear that all the poles are within the unit circle. Thus the closed-loop stability is ensured.

Table 7.2: RST controller for X and Y axis and $E(q^{-1})$ for disturbance estimation.

X axis	q^0	q^{-1}	q^{-2}	q^{-3}	q^{-4}
$R(q^{-1})$	0.32750694	−0.22567971	0.08539690		
$T(q^{-1})$	1.00000000	−1.29214711	0.47937122		
$S(q^{-1})$	0.00708911	0.00697230	0.00525065	0.00172280	0.00038752
$E(q^{-1})$	1.00000000	0.50540799	0.33664243		
Y axis	q^0	q^{-1}	q^{-2}	q^{-3}	q^{-4}
$R(q^{-1})$	0.26592206	−0.10131404	0.02261612		
$T(q^{-1})$	1.00000000	−1.29214711	0.47937122		
$S(q^{-1})$	0.00790339	0.00748054	0.00304940	0.00115195	0.00002851
$E(q^{-1})$	1.00000000	0.23728234	0.19946391		

7.7 Experimental results

The proposed ISMC was tested on the mini-CNC machine. The ISMC was designed and tested via a step-by-step approach starting from a RST controller with the independent objectives for regulation and tracking as described in the earlier sections. The controllers are designed based on the models that were obtained by the closed-loop identification method [31] presented in the Appendix.

7.7.1 RST controller

With the choice of P which was discussed in Section 7.5, the RST controllers for both the X and the Y axes are listed in Table 7.2.

With RST controllers used for both the X and the Y axes , a linear contour was generated with a feedrate of 40 mm/s and the results are shown in Fig. 7.5 (a). It is observed that the actual trajectory deviates from the desired linear contour on one side, thus resulting in a constant contour error with a magnitude of 6 μm at the steady state.

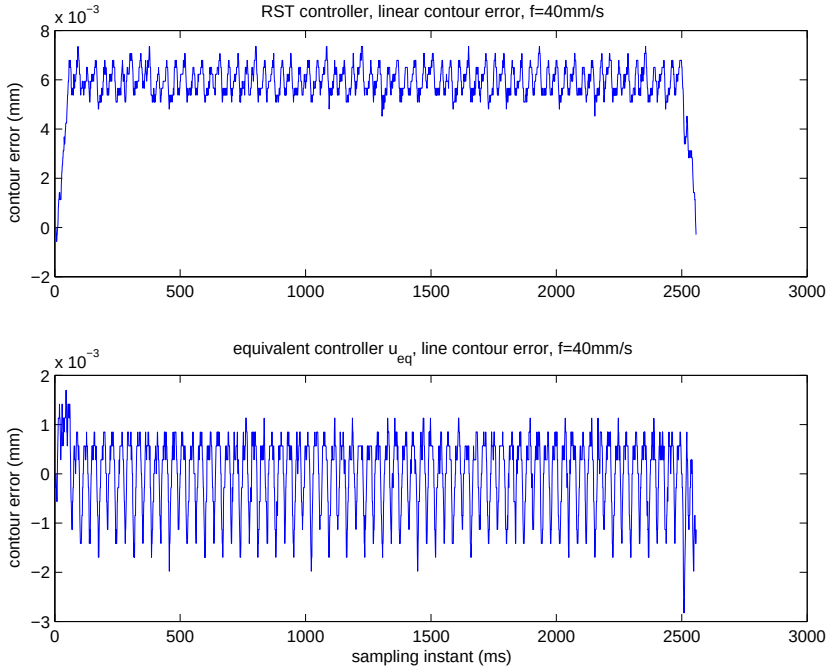

Figure 7.5: Linear contour error when f=40mm/s, using (a) RST controller, (b) equivalent control u_{eq}

A circular contour was also generated with a feedrate of $f = 40$ mm/s and the results are shown in Fig. 7.6 (a). For the RST controller, the individual axial tracking error cannot be reduced, and the profile of the resulting contour error exhibits a sinusoid variation with the maximum contour error at about 7 μm. It is also noticed from Fig. 7.6 (a) that there are spikes in the contour errors at the quadrant positions where static friction occurs when moving from standstill or at a reversal of velocities on one of the axes.

It is clear that the RST controller cannot significantly reduce the tracking errors, and thus indirectly reduce the contour errors.

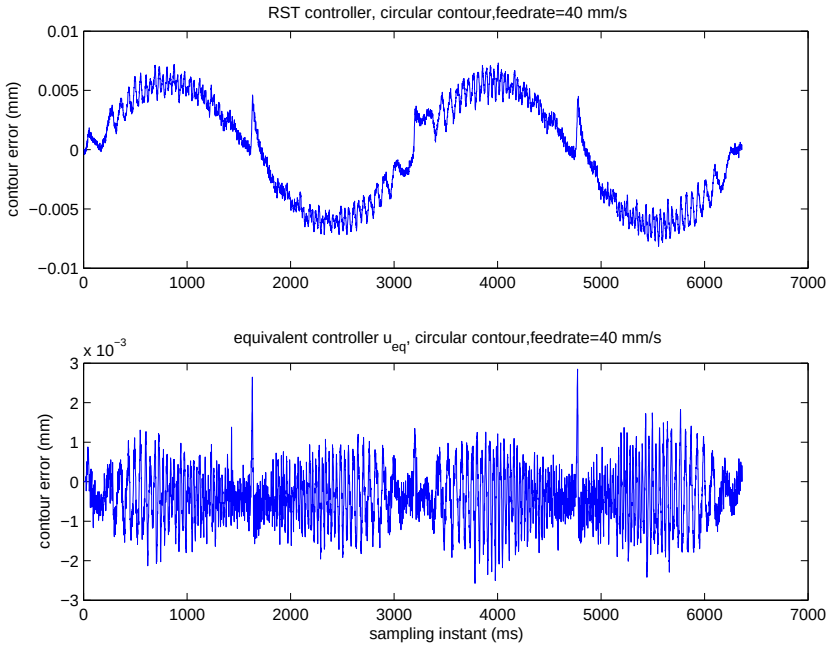

Figure 7.6: circular contour error when f=20mm/s, (a) RST controller, (b) equivalent control u_{eq}

7.7.2 Equivalent control

As derived in the previous sections, with the estimate of disturbance, the equivalent controller is derived from the expression of $s_k = 0$. The difference between the RST controller and the equivalent controller is that an disturbance estimate is included in the expression of equivalent control.

The equivalent control was first used to generate a linear contour with a feedrate of 40 mm/s and the results are shown in Fig. 7.5 (b). It is noted, compared with results for the RST controller, that the contour errors are much reduced and are now within a band of $\pm$ 2 μm except during the transient period.

A circular contour was then generated with the same feedrate of 40 mm/s and the

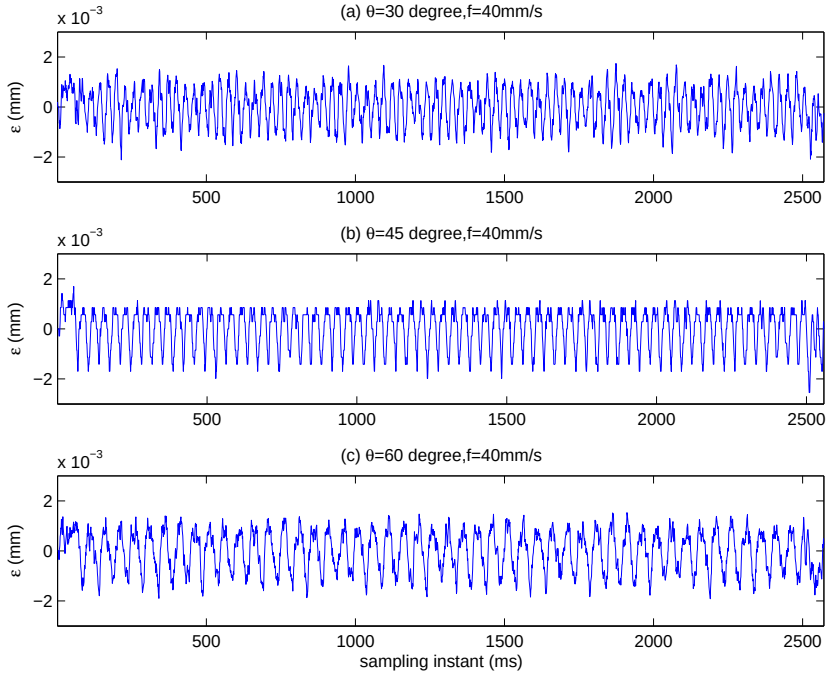

Figure 7.7: Linear contour error for integral sliding mode control: (a) $\theta = 30°$, (b) $\theta = 45°$ and (c) $\theta = 60°$

results are shown in Fig. 7.6 (b). Here again, it can be noted that with the equivalent controller, the contour errors are better than that for the RST controller and are now within a band of ± 3 μm. The profile of the contour errors now becomes flat. The stiction-induced spikes at the quadrant positions are also almost reduced by half. It can thus be seen that , with the disturbance estimate, the stiction torque as a form of external disturbance has been adequately incorporated into the control signal. It is thus concluded that, with equivalent control, the contour accuracy is greatly improved by the incorporation of the estimate of the disturbance.

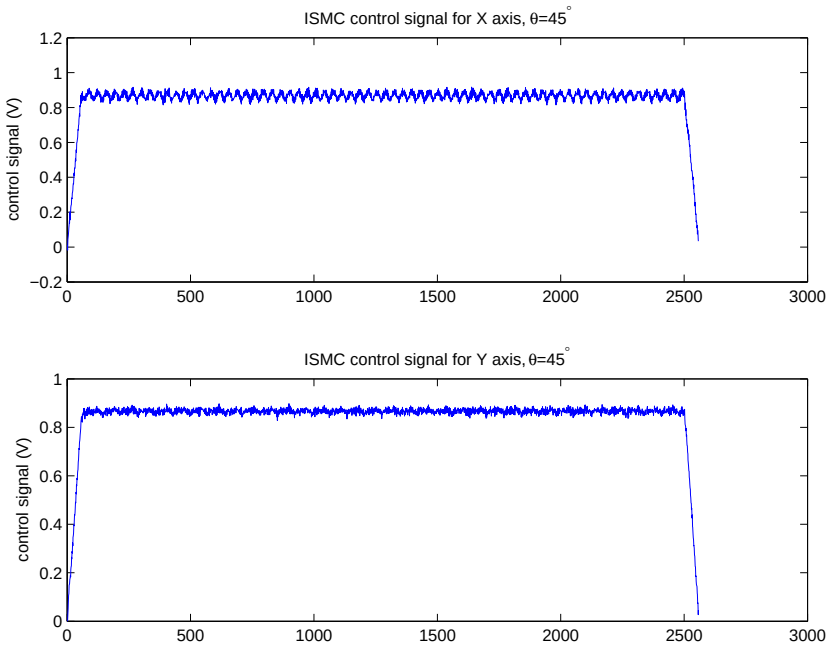

Figure 7.8: Control signal for integral sliding mode control when $\theta = 45°$

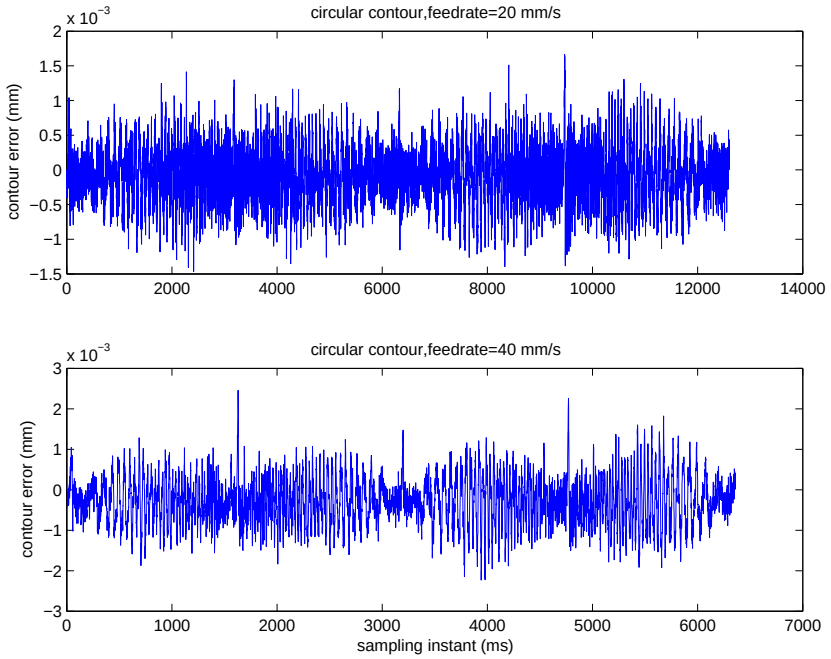

Figure 7.9: Integral SMC for circular contour: (a)f=20mm/s, (b)f=40mm/s

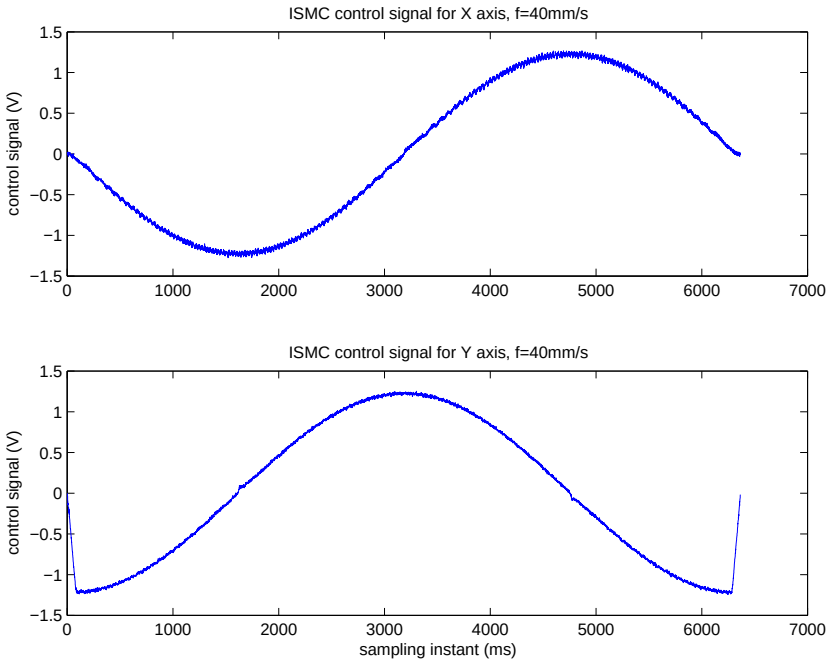

Figure 7.10: ISMC Control signal for circular contour when $f = 40$ mm/s

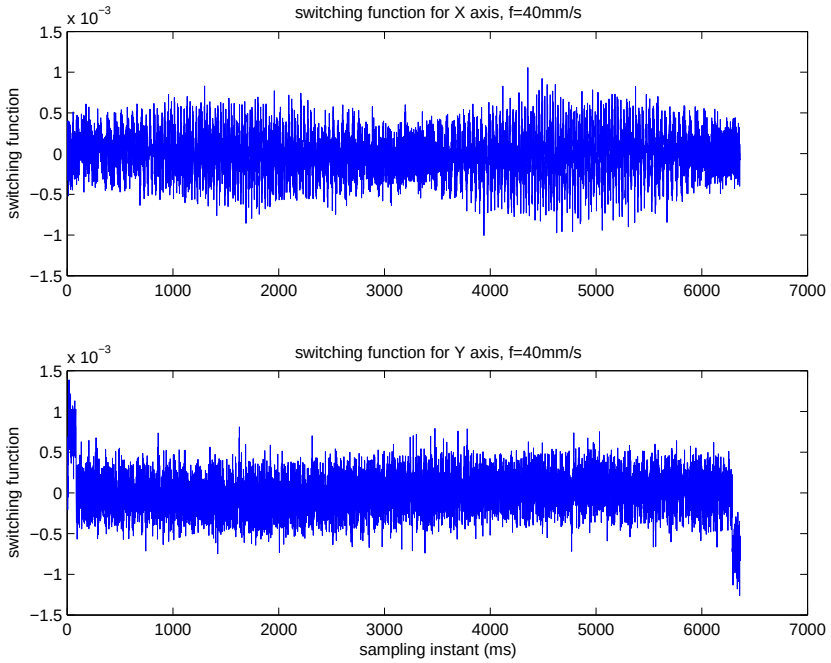

Figure 7.11: Switching function value for Integral SMC when running circular contour: (a)X axis, (b)Y axis

7.7.3 Integral sliding mode control

The proposed ISMC is used to run three linear contours whose angles with respect to the X axis are 30, 45 and 60°, respectively. The results, shown in Fig. 7.7, indicate that the contour errors are all within a band of $\pm$ 2 μm when the feedrate is 40 mm/s. This shows similar results as that for equivalent control in the steady state, but shows a slight improvement over equivalent control during the transient period. The ISMC control signals for both the X and the Y axes for the case when the feedrate is 40 mm/s and the angle with respect to the X axis is 45° are shown in Fig. 7.8. It can be noted that the control signals do not have undesirable high-frequency components causing chattering, which is good for the actuators.

The ISMC was next used to generate two circular contours with feedrates of 20 and 40 mm/s, respectively. The contour errors are shown in Fig. 7.9 with $k_i = 2.0$. Comparing Fig. 7.9 (b) with Fig. 7.6 (b) when $f = 40$ mm/s, it is observed that the contour error is slightly reduced although not significantly. The control signals for the X and the Y axes are shown in Fig. 7.10. Here again, it is noted that the control signals also do not contain undesirable high-frequency components.

The values of the switching function for both the X and the Y axes shown in Fig. 7.11 are within a band of 1.5 μm, and the mean values of the switching function are almost zero. The experimental results demonstrate that the integral switching function can drive the system into a small band around the defined sliding surface .

Another interesting observation is that the quadrant spikes shown in Fig. 7.6 (a) which are induced by stiction at the reversal of velocities or when starting from standstill have also been reduced even though there is no adoption of any stiction compensation

scheme [38].

Compared with two-degree-of-freedom (RST) control, the introduction of an estimate of disturbance to the equivalent controller can significantly reduce the contour error, while the introduction of integral control as the switching control signal can make sure the the system is finally driven onto the sliding surface, thus further improving the contouring accuracy.

7.8 Conclusion

In this chapter, an integral sliding mode controller (ISMC), based on input-output models, is proposed as a refinement from a two-degree-of-freedom controller with independent objectives for tracking and regulation. Thus, the knowledge of pole placement can be utilized in ISMC.

The robustness was improved by including a disturbance estimate, which results in the equivalent control. The estimate of the disturbance is made based on the assumption that the moving average of the disturbance does not change from the past d sampling instants to the future d instants. The use of the disturbance estimate has been shown, experimentally, to be able to significantly reduce the tracking errors, and thereby, the contour errors.

To eliminate the problem of chattering, two measures adopted were an appropriate choice of the sliding surface and an integral switching function. It was found that the choice of a natural frequency of the sliding surface dynamics that is slower than that of the open-loop transfer function can ease the problem of chattering. The choice of k_i, the integral switching gain, is based on the two-degree-of-freedom controller.

The root locus technique was used to determine an appropriate value of k_i to ensure closed-loop stability and good response.

The proposed ISMC was implemented and experimentally tested in a CNC machine. The contouring accuracy of the CNC machine was greatly improved by the proposed ISMC. Furthermore, no chattering was observed, which is beneficial to the machine actuators.

It is quite clear that different control strategies are closely related although they may have different terminologies. This design described here enables a step-by-step procedure to be carried out. Such an approach enables experimental tests to be carried out to fine tune the intermediate control action at each step, hence, making such a design procedure practical.

Chapter 8

Conclusion

8.1 Four methods of improving contouring accuracy

With the increasing demand on the dimensional accuracy of machined parts, much work has gone, and are going, into seeking various methods to improve the machining accuracy of CNC machines. Contouring accuracy, in terms of contour error, is a big concern for the designers and end-users of contouring (or continuous-path) type of CNC machines. Methods for achieving improved contouring accuracy can be classified as improving control of the machine's positioning axes to reduce axial tracking errors or directly reducing contour errors, or compensation approaches which compensates for these contour errors by either modifying the reference position inputs or other means.

Several approaches have been investigated in this thesis to improve contouring accuracy in CNC machines. Depending on their nature, these methods cover both control

and compensation.

One approach investigated is the use of simple position feedback controllers and tuning the proportional gains to match the axial dynamics by measuring the steady-state axial tracking errors. It is shown that this model-free approach of properly matching axial dynamics can achieve zero, or near-zero errors for linear contours. It also leads to good contouring accuracies for circular contours. With the addition of feedforward compensation gain at the reference position inputs, the radial errors resulting from the limited bandwidth of the axial dynamics can be further reduced to negligible values.

With matched axial dynamics and feedforward compensation for radial errors, near-perfect linear and circular contours can be achieved. The main source of contour errors is then stiction and other nonlinear effects. These cause significant contour error spikes at the quadrant positions in circular contours when there is a velocity reversal or when moving from standstill. A two-stage stiction compensation scheme is then proposed, and further investigated for its performance, in this thesis to take care of stiction. Experiments carried out on a mini-CNC machine show the good performance of this compensation scheme with the contour spikes at the quadrant positions significantly reduced such that their effect is no longer pronounced.

A model-based Taylor series expansion error compensation (TSEEC) approach, which formulates the contour error as a Taylor series expansion around points along the reference path and with the compensation components as deviations from these reference points, is also proposed, developed and investigated. Through simulation studies, it

Table 8.1: Comparison of four methods

	Models	Implementation	linear or circular	performance
Tuning gain	No	Easy	Both	good
Stiction compensation	No	Easy	circular	good
TSEEC	Yes	Difficult	both	fair
ISMC	Yes	Difficult	both	very good

was shown that with perfect knowledge of the axial dynamics, this TSEEC compensation approach can achieve perfect contouring with zero contour errors for linear and circular contours. Experimental evaluation of this approach, implemented on a mini-CNC machine and using discrete-time dynamic models obtained from experimental identification, also show very good contouring performance.

The application of an integral sliding mode control (ISMC)approach, known for its robustness against model uncertainties and external disturbances, is also investigated for it performance in achieving good contouring accuracy. Here, a step-by-step approach is used in the design of the ISMC controller. Experimental results show that the ISMC cpproach can improve the contouring accuracy greatly, even at the quadrant positions where stiction and other non-linear effects causes error spikes due to the reversal of velocities and when starting from standstill.

For an easy overview of these four methods, the advantages and disadvantages of these methods are compared in Table 8.1.

8.2 Contributions of this thesis

This thesis has made contributions in proposing, developing and investigating the following approaches for achieving improved contouring accuracy in CNC machines. Simulation studies and actual implementation on a mini-CNC machine show that all

these approaches can reduce contour errors very significantly, with errors reduced to levels comparable to those caused by vibrations in the machine during motion.

1. A simple and straightforward approach, using simple proportional controller, to match axial dynamics by tuning control loop gains and a feedforward compensation approach to further reduce the radial contour error due to the limited bandwidth of the axial dynamics;

2. A two-stage stiction compensation scheme to compensate for the contour error spikes at the quadrant positions in circular contours.

3. A model-based Taylor series expansion error compensation (TSEEC) approach which has been shown to be able to achieve perfect contouring with no contour errors when perfect knowledge of the machine's dynamic model is available. Due to inevitable modeling errors and other disturbance effects, the TSEEC approach needs to be modified to achieve good results. Several such modification are proposed and investigated and shown to work well, achieving very good contouring performance. These modifications include a) making the compensation target as making the moving average of the contour errors, rather than the instantaneous value, to be zero, b) the inclusion of low-pass filters to filter out the high-frequency components in the compensation signals, and c) a compensation gain which can be adjusted for optimal performance.

4. A systematic step-by-step design for an integral sliding mode controller (ISMC) which can achieve good contouring accuracies. The sliding surface is designed such that the normally encountered chattering problem is avoided. It is also shown that the inclusion of integral control action, with the integral gain chosen

with the help of the root locus, can further improve contouring accuracy even at the quadrant positions where error spikes occur due to stiction.

8.3 Possible future research topics

Several approaches have been proposed to improve contouring accuracy in CNC machines. Achieving ever better contouring accuracies in CNC machines will continue to be a topic of interest as demand for ever accurate machined parts continue. Some possible topics for further research are briefly described below.

The first concerns stability consideration in the proposed TSEEC method. Because TSEEC generates its compensation signals based on the position feedback and the identified model, it is a feedback control system and, as such, guaranteed stability becomes a concern. Research could be done to investigate how stability in the feedback TSEEC system can be guaranteed or best achieved.

The second possible area for further investigation involves an extension to TSEEC to make it applicable for arbitrary curves other than just linear and circular contours. One possible approach is to regard any arbitrary contour as a series of small linear contours for which TSEEC is applicable. The problem then reduces to investigating how TSEEC can be applied to the discontinuity at the transition between one small straight segment and the next. Another approach would be to make use of the states of the immediate past sampling instants to predict the error for the immediate future sampling instances and to develop compensating signals based on these.

In the design of integral sliding mode control (ISMC), the choice of the closed-loop

poles were discussed in details. Further studies into the convergence of the dynamics onto the predefined sliding surface still needs to be done so that this can be guaranteed under specified conditions.

In the work described here, the performance of ISMC was evaluated under conditions of no cutting. It is useful to test the performance of ISMC under various cutting conditions. Cutting forces can be considered as external disturbances and how well and robust ISMC is to this still needs to be studied.

Bibliography

[1] *Enhanced machine controller, http://www.linuxcnc.org.*

[2] Y. Altintas, K. Erkorkmaz, and W.-H Zhu, *Sliding mode controller design for high speed feed drives*, Annals of the CIRP **49** (2000), no. 1, 265–270.

[3] Brian Armstrong-Hélouvry, *Control of machines with friction*, Kluwer Academic Publishers, Norwell, MA, 1991.

[4] Brian Armstrong-Hélouvry, Pierre Dupont, and Carlos Canudas De Wit, *A survey of models, analysis tools and compensation methods for the control of machines with friction*, Automatica **30** (1994), no. 7, 1083–1138.

[5] Karl J. Åström and Björn Wittenmark, *Computer-controlled systems: theory and design*, 3rd ed., Prentice Hall, Upper Saddle River, NJ, 1997.

[6] Alessandro Beghi, Lorenzo Nardo, and Marco Stevanato, *Obsever-based discrete-time sliding mode throttle control for drive-by-wire operation of a racing motorcycle engine*, IEEE Transactions on Control Systems Technology **14** (2006), no. 4, 767–775.

[7] C. L. Chen, M. J. Jang, and K. C. Lin, *Modeling and high-precision control of a ball-screw-driven stage*, Precesion Engineering **28** (2004), no. 4, 483–495.

[8] Shyh-Leh Chen, Hung-Liang Liu, and Sing Ching Ting, *Contouring control of biaxial systems based on polar coordinates*, IEEE/ASME Transactions on Mechatronics **7** (2002), no. 3, 329–345.

[9] Tzuen-Lih Chern, Chin-Wen Chuang, and Rong-Lin Jiang, *Design of discrete-time integral variable structure control systems and application to a brushless dc motor control*, Automatica **32** (1996), no. 5, 773–779.

[10] Jih-Hua Chin, Yuan-Ming Cheng, and Jin-Huei Lin, *Improving contour accuracy by fuzzy-logic enhanced cross-coupled precompensation method*, Robotics and Computer-Integrated Manufacturing **20** (2004), 65–76.

[11] Jih-Hua Chin and Tsung-Ching Lin, *Cross-coupled precompensation method for the contouring accuracy of computer numerically controlled machine tools*, International Journal of Machine Tools and Manufacture **37** (1997), no. 7, 947–967.

[12] Jih-Hua Chin and Her-Chyi Tsai, *A path algorithm for robotic machining*, Robotics and Computer-Integrated Manufacturing **10** (1993), no. 3, 185–198.

[13] George T.-C. Chiu and Masayoshi Tomizuka, *Contouring control of machine tool feed drive systems: a task coordinate frame approach*, IEEE Transactions on Control Systems Technology **9** (2001), no. 1, 130–139.

[14] Hua-Yi Chuang and Chang-Huan Liu, *Cross-coupled adaptive feedrate control for multiaxis machine tools*, Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME **113** (1991), 451–457.

[15] ———, *A model-referenced adaptive control strategy for improving contour accuracy of multiaxis*, IEEE Transactions on Industry Applications **28** (1992), no. 1, 221–227.

[16] S. V. Drakunov and V. I. Utkin, *Sliding mode in dynamic systems*, International Journal of Control **55** (1990), 1029–1037.

[17] J. Francis Eeintjes, *Numerical control: making a new technology*, Oxford University Press, New York, 1991.

[18] Yongsoon Eun, Jung-Ho Kim, Kwangsoo Kim, and Dong-il Cho, *Discrete-time variable structure controller with a decoupled disturbance compensator and its application to a cnc servomechanism*, IEEE Transactions on Control Systems Technology **7** (1999), no. 4, 414–423.

[19] W. Gao, Y. Wang, and A. Homaifa, *Discrete-time variable structure control systems*, IEEE Transactions on Industrial Electronics **42** (1995), no. 2, 117–122.

[20] B. Haack and Masayoshi Tomizuka, *The effect of adding zeros to feedforward controllers*, Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME **113** (1991), 6–10.

[21] Hsin-Chiang Ho, Jia-Yush Yen, and Shui-Shong Lu, *A decoupled path-following control algorithm based upon the decoupled trajectory error*, International Journal of Machine Tools and Manufacture **39** (1999), 1619–1630.

[22] Seong-Wook Hong, Young-Jae Shin, and Hu-Sang Lee, *An efficient method for identification of motion error sources from circular test results in nc machines*, International Journal of Machine Tools and Manufacture **37** (1997), no. 3, 327–340.

[23] Makoto Iwasaki, Tomohiro Shibata, and Nobuyuki Matsui, *Disturbance-observer-based nonlinear friction compensation in table drive system*, IEEE Transactions on Mechatronics **4** (1999), no. 1, 3–8.

[24] Makoto Iwasaki, Hiroto Takei, and Nobuyuki Matsui, *GMDH-based modeling and feedforward compensation for nonlinear friction in table drive systems*, IEEE Transactions on Industrial Electronics **50** (2003), no. 6, 1172–1178.

[25] Dong-Il Kim and Sungkwun Kim, *An iterative learning control method with application for cnc machine tools*, IEEE Transactions on Industry Applications **32** (1996), no. 1, 66–72.

[26] Jung-Ho Kim, Seung-Hyun Oh, Dong-il Cho, and J. Karl Hedrick, *Robust discrete-time variable structure control methods*, Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME **122** (2000), no. 4, 766–775.

[27] Yoram Koren, *Cross-coupled biaxial computer control for manufacturing systems*, Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME **102** (1980), no. 4, 265–272.

[28] ———, *Computer control of manufacturing systems*, McGraw-Hill, New York, 1983.

[29] Yoram Koren and Chin-Ching Lo, *Variable-gain cross-coupling controller for contouring*, Annals of the CIRP **40** (1991), no. 1, 371–374.

[30] ———, *Advanced controllers for feed drives*, Annals of the CIRP **41** (1992), no. 2, 689–698.

[31] D. Ioan Landau and Gianluca Zito, *Digital control systems: design, identification and implementation*, Springer-Verlag, London, 2006.

[32] Yu-Feng Li and Jan Wikander, *Model reference discrete-time sliding mode control of linear motor precision servo systems*, Mechatronics **14** (2004), 835–851.

[33] Chin-Ching Lo and Ching-Yei Chung, *Tangential-contouring controller for biaxial motion control*, Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME **121** (1999), 126–129.

[34] Chin-Ching Lo and Chao-Yin Hsiao, *Cnc machine tool interpolator with path compensation for repeated contour machining*, Computer-Aided Design **30** (1998), no. 1, 55–62.

[35] ———, *A method of tool path compensation for repeated machining process*, International Journal of Machine Tools and Manufacture **38** (1998), no. 3, 205–213.

[36] O. Masory, *Improving contouring accuracy of nc/cnc systems with additional velocity feed forward loop*, Journal of Engineering for Industry, Transactions ASME **108** (1986), no. 3, 227–230.

[37] Xuesong Mei, Maosaomi Tsutsimi, Takanori Yamazaki, and Nuogang Sun, *Study of the friction eror for a high-speed high precision table*, International Journal of Machine Tools and Manufacture **41** (2001), 1405–1415.

[38] Xuesong Mei, Masaomi Tsutsumi, Tao Tao, and Nuogang Sun, *Study on the compensation of error by stick-slip for high-precision table*, International Journal of Machine Tools and Manufacture **44** (2004), no. 5, 503–510.

[39] Katsuhiko Ogata, *Modern control engineering*, 4th ed., Prentice-Hall, Inc, Upper Saddle River, NJ, 2002.

[40] K. D. Oldknow and I. Yellowley, *Three-dimensional dynamic interpolation using stateline based control architecture*, International Journal of Machine Tools and Manufacture **42** (2002), 1627–1641.

[41] ———, *Implementation and validation of 3-dimensional dynamic interpolation using an fpga based controller*, International Journal of Machine Tools and Manufacture **43** (2003), 937–945.

[42] Eun-Chan Park, Hyuk Lim, and Chong-Ho Choi, *Position control of x-y table at velocity reversal using presliding friction characteristics*, IEEE Transactions on Control Systems Technology **11** (2003), no. 1, 24–31.

[43] Aun-Neow Poo, John G. Bollinger, and George W. Younkin, *Dynamic errors in type 1 contouring systems*, IEEE Transactions on Industry Applications **IA-8** (1972), no. 4, 477–484.

[44] R. Ramesh, M. A. Mannan, and A. N. Poo, *Error compensation in machine tools - a review. part I: Geometric, cutting-force induced and fixture-dependent errors*, International Journal of Machine Tools and Manufacture **40** (2000), no. 9, 1235–1256.

[45] ———, *Error compensation in machine tools - a review. part II: Thermal errors*, International Journal of Machine Tools and Manufacture **40** (2000), no. 9, 1257–1284.

[46] ———, *Tracking and contour error control in cnc servo systems*, International Journal of Machine Tools and Manufacture **45** (2005), no. 3, 301–326.

[47] Ramakrishnan Ramesh, *Measurement and modelling of machine tool thermal error under varying operating conditions*, Ph.D. thesis, National University of Singapore, 2002.

[48] R. J. Seethaler and I. Yellowley, *The regulation of position error in contouring systems*, International Journal of Machine Tools and Manufacture **36** (1996), no. 6, 713–728.

[49] Yi-Ti Shih, Chin-Sheng Chen, and An-Chen Lee, *A novel cross-coupling control design for bi-axis motion*, International Journal of Machine Tools and Manufacture **42** (2002), 1539–1548.

[50] Jean-Jacques E. Slotine and Weiping Li, *Applied nonlinear control*, Prentice Hall, Englewood Cliffs, N.J., 1991.

[51] K. Srinivasan and P. K. Kulkarni, *Cross-coupled control of biaxial feed drive servomechanisms*, Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME **112** (1990), 225–232.

[52] Mingxuan Sun, Youyi Wang, and Danwei Wang, *Variable -structure repetitive control: a discrete-time strategy*, IEEE Transactions on Industrial Electronics **53** (2005), no. 2, 610–616.

[53] Y. S Tarng and H. E. Cheng, *An investigation of stick-slip friction on the contouring accuracy of cnc machine tools*, International Journal of Machine Tools and Manufacture **35** (1995), no. 4, 565–576.

[54] Y. S Tarng, Hua-Yi Chuang, and W. T. Hsu, *An optimization approach to the contour error control of cnc machine tools using genetic algorithms*, International Journal of Advanced manufacturing Technology **13** (1997), 359–366.

[55] Y. S Tarng, J. Y. Kao, and Y. S. Lin, *Identification of and compensation for backlash on the contouring accuracy of cnc machinig centres*, International Journal of Advanced manufacturing Technology (1997), no. 13, 77–85.

[56] Masayoshi Tomizuka, *Zero phase error tracking algorithm for digital control*, Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME **109** (1987), 65–68.

[57] Masayoshi Tomizuka, Jwu-Sheng Hu, Tsu-Chih Chiu, and Takuya Kamano, *Synchronization of two motion control axes under adaptive feedforward control*, Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME **114** (1992), 196–203.

[58] D. Torfs, J. De Schutter, and J. Swevers, *Extended bandwidth zero phase error tracking control of nonminimal phase systems*, Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME **114** (1992), 347–351.

[59] M.-C Tsai, I.-F. Chiu, and M.-Y Cheng, *Design and implementation of command and friction feedforward control for cnc motion controllers*, IEE proceedings Control Theory and Application **151** (2004), no. 1, 13–20.

[60] V. I. Utkin, *Sliding modes in control optimization*, Springer, New York, 1992.

[61] Ruh-Hua Wu and Pi-Cheng Tung, *Studies of stick-slip friction, presliding displacement, and hunting*, Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME **124** (2002), no. 1, 111–117.

[62] Xue-Cheng Xi, Aun-Neow Poo, and Siaw-Kiang Chou, *Factors affecting contour errors in CNC systems*, Proceedings of 2005 9th International conference on mechatronics technology (Kuala Lumpur, Maylasia), 2005, pp. ICMT–117.

[63] Longxiang Yang and Ian Yellowley, *High-speed contouring using a novel dynamic interpolation mechanism*, International Journal of Machine Tools and Manufacture **41** (2001), 773–794.

[64] X. Ye, X. Chen, X. Li, and S. Huang, *A cross-coupled path precompensation algorithm for rapid prototyping and manufacturing*, International Journal of Advanced manufacturing Technology **20** (2002), 39–43.

[65] Syh-Shiuh Yeh and Pau-Lo Hsu, *Analysis and design of the integrated controller for precise motion systems*, IEEE Transactions on Control Systems Technology **7** (1999), no. 6, 706–717.

[66] ———, *Theory and applications of the robust cross-coupled control design*, Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME **121** (1999), 524–530.

[67] ———, *Estimation of the contouring error vector for the cross-coupled control design*, IEEE/ASME Transactions on Mechatronics **7** (2002), no. 1, 44–51.

[68] ———, *Analysis and design of integrated control for multi-axis motion systems*, IEEE Transactions on Control Systems Technology **11** (2003), no. 3, 375–382.

[69] ———, *Perfectly matched feedback control and its integrated design for multi-axis motion systems*, Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME **126** (2004), no. 3, 547–557.

[70] Q. Zhong, Y. Shi, J. Mo, and S. Huang, *A linear cross-coupled control system for high-speed machining*, International Journal of Advanced Manufacturing Technology **19** (2002), 558–563.

Appendix A

Closed-loop identification

A.1 Introduction

The Taylor series expansion error compensation (TSEEC) and the integral sliding mode control (ISMC) are the two proposed model-based methods for reducing contour errors. The TSEEC scheme is based on the discrete-time model of the closed-loop axial drive system while the ISMC is based on the discrete-time model of the open loop axial drive system which includes the velocity loop and the integrator as shown in Fig. A.1.

For these two model-based algorithms, the discrete-time models of the axial dynamics, identified from the input-output data, are thus needed. As can be seen from Chapter 4, the axial drive consists of an inner velocity loop and an outer position loop. The inner velocity loop, built into the servo motor drive that was used in this project, has a PI controller. As the closed loop model of the axial drive can be easily derived from the open loop model, we will mainly focus on the identification of the open loop model as shown in the enclosed part by dotted line in Fig. A.1.

In the situation where the open loop system is stable and there is no integrator, the model of the system can be obtained by the input-output data in open loop.

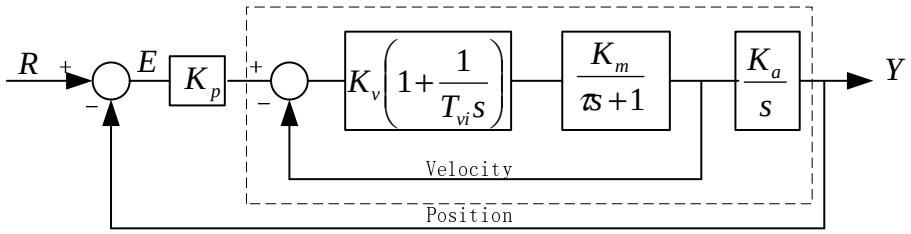

Figure A.1: Closed-loop axial dynamics with open loop enclosed in dotted line.

The existence of drifting and integrator, however, make it difficult to apply the open loop identification to the axial drive system. It is thus more convenient to make the servo motor under closed-loop control so as to stop the drifting. With the feedback controller known *a priori*, the axial dynamic model can be obtained by the method of closed-loop identification, which was thoroughly discussed in [31].

A.2 Discrete-time input-Output model

The axial drive system can be expressed in the discrete-time input-output model

$$A(q^{-1})y(k) = q^{-d}B(q^{-1})u(k) + w(k) \tag{A-1}$$

where d is the time delay which is a multiple of the sampling interval T_s. $w(k)$ is the lumped disturbance which includes the external disturbance and model uncertainties. The numerator $B(q^{-1})$ and the denominator $A(q^{-1})$ are of order m and n respectively,

$$B(q^{-1}) = b_0 + b_1q^{-1} + b_2q^{-q} + \cdots + b_mq^{-m} \tag{A-2}$$

$$A(q^{-1}) = 1 + a_1q^{-1} + a_2q^{-2} + \cdots + a_nq^{-n} \tag{A-3}$$

The objective of the identification is to obtain the coefficients of $B(q^{-1})$ and $A(q^{-1})$.

The model output can be also written as a dot product between the unknown parameter vector $\theta(k)$ and the plant model regressor vector $\phi(k)$

$$y(k) = \theta(k)^T\phi(k) \tag{A-4}$$

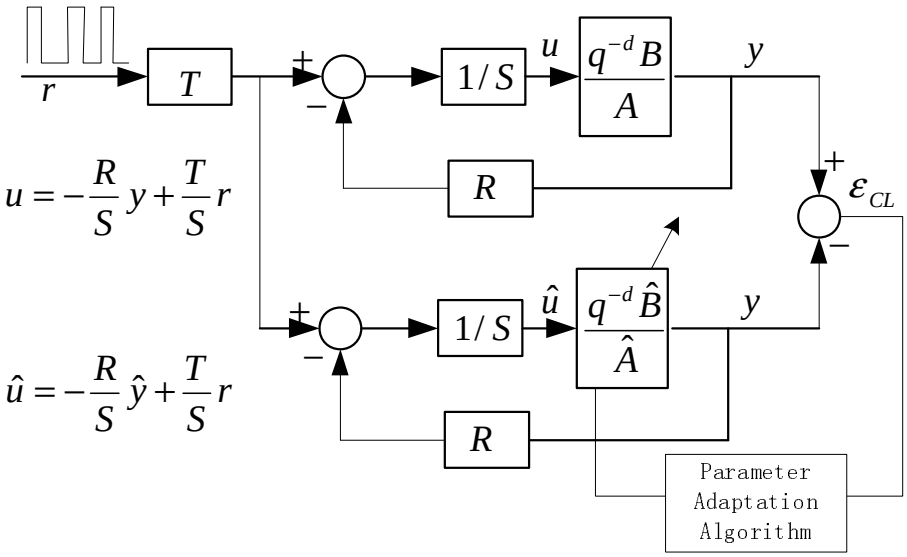

Figure A.2: Identification in closed loop (excitation added to reference).

A.3 Closed loop identification

The principle of closed loop identification is illustrated in Fig. A.2. Basically, there are two positions where the excitation signals are fed into the closed loop. One is at the reference input which is shown in Fig. A.2, and the other is at the normal control signal. In the later case, the excitation signals are superimposed onto the normal control signals. Both schemes are quite similar and the same parameter adaption algorithms are used. In this project, the excitation signals are applied at the reference input.

The prediction error between the output of the real closed loop system and the closed loop predictor (closed loop output error) is the measure of the difference between the true plant and the estimated one. This error can be used to adapt the estimated model such that the error is minimized.

A.3.1 The CLOE, F-CLOE and AF-CLOE method

Based on the different choice of the regression vector $\phi(k)$, there are a few variants of closed-loop identification methods. More details are given in [31].

Closed loop output error(CLOE)

$$y(k+1) = \theta^T \phi(k) \tag{A-1}$$

$$u(k) = -\frac{R(q^{-1})}{S(q^{-1})} y(k) + r_u(k) \tag{A-2}$$

where θ is the true parameters of the plant. The adjustable closed-loop predictor is described by

$$\hat{y}^o(k+1) = \hat{\theta}(k)^T \phi(k) \tag{A-3}$$

$$\hat{y}(k+1) = \hat{\theta}(k+1)^T \phi(k) \tag{A-4}$$

$$\hat{u}(k) = -\frac{R(q^{-1})}{S(q^{-1})} \hat{y}(k) + r_u(k) \tag{A-5}$$

where $\hat{y}^o(k+1)$ and $\hat{y}(k+1)$ denote the *a priori* and the *a posteriori* outputs of the closed-loop predictor. $\hat{u}(k)$ is the control signal computed by the same controller of the plant. The difference between $u(k)$ and $\hat{u}(k)$ lies in the fact that $u(k)$ is calculated by the actual plant output $y(k)$ as in Eq. (A-2) while $\hat{u}(k)$ is based on the *a posteriori* outputs of the closed-loop predictor as in Eq. (A-5).

The closed-loop prediction error is then given by

$$\varepsilon^o_{CL}(k+1) = y(k+1) - \hat{y}^o(k+1) \qquad \textit{a priori} \tag{A-6}$$

$$\varepsilon_{CL}(k+1) = y(k+1) - \hat{y}(k+1) \qquad \textit{a posteriori} \tag{A-7}$$

The parameter adaption algorithm is listed

$$\begin{aligned}
\hat{\theta}(k+1) &= \hat{\theta}(k) + F(k)\Phi(k)\varepsilon_{CL}(k+1) && \text{(A-8)}\\
F(k+1) &= \lambda_1(k)F(k)^{-1} + \lambda_2(k)\Phi(k)\Phi(k)^T && \text{(A-9)}\\
& \quad 0 < \lambda_1(k) \leq 1; \quad 0 < \lambda_2(k) < 2 \\
\varepsilon_{CL}(k+1) &= \frac{\varepsilon^o_{CL}(k+1)}{1+\Phi(k)^T\Phi(k)} && \text{(A-10)}\\
\Phi(k) &= \phi(k) && \text{(A-11)}
\end{aligned}$$

The vector are expressed in

$$\begin{aligned}
\hat{\theta}(k)^T &= [\hat{a_1}(k), \cdots, \hat{a}_{n_A}(k), \hat{b}_1(k), \cdots, \hat{b}_{n_B}(k)] && \text{(A-12)}\\
\phi(k)^T &= [-\hat{y}(k), \cdots, -\hat{y}(k-n_A+1), && \text{(A-13)}\\
& \quad \hat{u}(k-d), \cdots, \hat{u}(k-d-n_B+1)]
\end{aligned}$$

The convergence of CLOE, in the absence of noise, is subject to a sufficient condition that the transfer function

$$\frac{S(q^{-1})}{P(q^{-1})} - \frac{\lambda_2}{2}; \qquad 2 > \lambda_2 \geq \max \lambda_2(k) \tag{A-14}$$

should be strictly positive real, where the polynomial

$$P(q^{-1}) = A(q^{-1})S(q^{-1}) + q^{-d}B(q^{-1})R(q^{-1}) \tag{A-15}$$

defines the poles of the closed loop. A strictly positive real transfer function [31, pp. 250] is characterized by the following two properties:

1. the transfer function is asymptotically stable;
2. the real part of the transfer function is positive for all frequencies.

Filtered closed loop output error(F-CLOE)

To relax the strictly positive real condition, the vector $\phi(k)$ is filtered by $S(q^{-1})/\hat{P}(q^{-1})$, where $\hat{P}(q^{-1})$ is an estimation of the closed loop polynomial defined as

$$\hat{P}(q^{-1}) = \hat{A}(q^{-1})S(q^{-1}) + q^{-d}\hat{B}(q^{-1})R(q^{-1}) \tag{A-16}$$

Correspondingly in the parameter adaptation algorithm, we use

$$\Phi(k) = \frac{S(q^{-1})}{\hat{P}(q^{-1})}\phi(k) \tag{A-17}$$

Adaptive filtered closed loop output error(AF-CLOE)

Instead of using a fixed value $\hat{P}(q^{-1})$, we can use a time varying $\hat{P}(k, q^{-1})$ which is adaptively calculated by

$$\hat{P}(k, q^{-1}) = \hat{A}(k, q^{-1})S(q^{-1}) + q^{-d}\hat{B}(k, q^{-1})R(q^{-1}) \tag{A-18}$$

thus $\phi(k)$ is filtered by a time-varying filter

$$\Phi(k) = \frac{S(q^{-1})}{\hat{P}(k, q^{-1})}\phi(k) \tag{A-19}$$

Extended closed loop output error(X-CLOE)

If it is assumed that the disturbance acting on the plant output can be represented by an ARMAX model,

$$A(q^{-1})y(k) = q^{-d}B(q^{-1})u(k) + C(q^{-1})e(k) \tag{A-20}$$

$$u(k) = -\frac{R(q^{-1})}{S(q^{-1})}y(k) + r_u(k) \tag{A-21}$$

X-CLOE is an identification method based on the whitening of the closed loop prediction error. An adjustable closed loop predictor of the following form is used

$$\hat{y}^0(k+1) = -\hat{A}^*(k,q^{-1})\hat{y}(k) + \hat{B}^*(k,q^{-1})\hat{u}(k-d) + \hat{H}^*(k,q^{-1})\frac{\varepsilon_{CL}(k)}{S(q^{-1})} \quad \text{(A-22)}$$

$$= \hat{\theta}_e(k)^T\phi_e(k) \quad \text{(A-23)}$$

$$\hat{u}(k) = -\frac{R(q^{-1})}{S(q^{-1})} + r_u(k) \quad \text{(A-24)}$$

$$\hat{\theta}_e(k)^T = [\hat{a}_1, \cdots, \hat{a}_{n_A}(k), \hat{b}_1(k), \ldots, \hat{b}_{n_B}(k), \hat{h}_1^*(k), \ldots, \hat{h}_{n_H}(k)] \quad \text{(A-25)}$$

$$\phi_e(k)^T = [\phi(k)^T, \varepsilon_{CLf}(k), \ldots, \varepsilon_{CLf}(k-n_H+1)] \quad \text{(A-26)}$$

$$\varepsilon_{CLf} = \frac{\varepsilon_{CL}(k)}{S(q^{-1})} \quad \text{(A-27)}$$

$$\varepsilon^0_{CL}(k+1) = y(k+1) - \hat{y}^0(k+1) \quad \text{(A-28)}$$

$$\varepsilon_{CL}(k+1) = y(k+1) - \hat{\theta_e}^T(k+1)\phi_e(k) \quad \text{(A-29)}$$

The parameter adaption algorithm is given by Eq. (A-8) through (A-11) where

$$\hat{\theta}(k) = \hat{\theta}_e(k); \qquad \Phi(k) = \phi_e(k) \quad \text{(A-30)}$$

A.3.2 Pseudo-random Binary Sequences (PRBS)

Pseudo-random binary sequences are sequences of rectangular pulses, modulated in width, which approximate a discrete-time white noise and thus have a spectral content rich in frequency.

The name *pseudo-random* comes from the fact that they are characterized by a sequence length which the variations in pulse width vary randomly, but over a large time horizon, they are periodic. The period is defined by the length of the sequence.

A.4 Preprocessing the training data

Although the original input-output can directly be used to estimate the coefficients of the model, two additional measures can be taken to improve the quality of the identified model. One is to use the *a priori* information about the plant. For the open loop axial dynamics, there exists an integrator. Instead of identifying the original input-output relationship, we alternatively identify the relationship between the input and the variation of the output, i.e., $u(k)$ vs. $(y(k)-y(k-1))$. The second measure taken is to scale the input output data before doing parameter adaptation. This is because the parameter adaptation algorithm involves the calculation of the covariance matrix $(\Phi(k)\Phi(k)^T)$. If the magnitude of some elements of the vector $\Phi(k)$ is significantly larger than that of others, the elements with small magnitude will become less significant in the parameter adaptation. To make all the elements equally important in the parameter adaptation, we preprocess the data so that all the elements are in the same scale.

To incorporate the information of the existence of an integrator and to facilitate the scaling, the original data needs to be pre-processed and the obtained model needs to be post-processed. The original block diagram is redrew to obtain the equivalent diagram for easy understanding.

For a system with an integrator, the transfer function can be written as

$$\frac{y(k)}{u(k)} = \frac{q^{-d}B(q^{-1})}{A(q^{-1})} = \frac{q^{-d}B(q^{-1})}{A_1(q^{-1})(1-q^{-1})} \tag{A-1}$$

Denote $y'(k) = y(k)(1-q^{-1})$ as the variation of the output. The identification is to obtain the input-output relationship between $u(k)$ and $y'(k)$, i.e.:

$$\frac{y'(k)}{u(k)} = \frac{q^{-d}B(q^{-q})}{A_1(q^{-q})} \tag{A-2}$$

With this change in the identified output variable, the original block diagram needs to be modified accordingly as shown in Fig. A.3. An open-loop system without an

integrator is controlled by a RST controller as shown in Fig. A.3 (a). By moving out the integrator in the closed loop (Fig. A.3 (b)), and the equivalent controller is derived as $S' = S(1 - q^{-1})$ and $T' = T(1 - q^{-1})$ and R is kept unchanged. Thus, the input-output transfer function is the relationship between $u(k)$ and $y'(k)$. Secondly, to make variation of $u(k)$ and $y'(k)$ are in the same range, the new output variable $y'_\alpha(k)$ is formed by multiplying $y'(k)$ by α, where α is defined as

$$\alpha = \frac{u_{max} - u_{min}}{y'_{max} - y'_{min}} \tag{A-3}$$

where u_{max} and u_{min} and maximum and minimum value of $u(k)$ respectively. y'_{max} and y'_{min} are defined similarly.

After incorporating the information of an integrator, and scaling the input-output data, the transfer function is obtained between $u(k)$ and y'_α

$$\frac{y'_\alpha(k)}{u(k)} = \frac{q^{-d}B_1(q^{-1})}{A_1(q^{-1})} = \frac{q^{-d}\alpha B(q^{-1})}{A_1(q^{-1})} \tag{A-4}$$

Finally, the original transfer function $y(k)/u(k)$ is obtained after post-processing Eq. (A-4)

$$\frac{y(k)}{u(k)} = \frac{q^{-d}B(q^{-1})}{A(q^{-1})} = \frac{q^{-d}B_1(q^{-1})/\alpha}{A_1(q^{-1})(1 - q^{-1})} \tag{A-5}$$

A.5 Validation of models

The input-output data is collected after the PRBS signals are applied at the reference input of the system. The coefficients can be obtained by the aforementioned four different methods, i.e., CLOE, F-CLOE, AF-CLOE and X-CLOE. All these four methods can be used. However, there is no one method that can be used for all cases. The practical way is to use the validation to choose one method that can give the best statistical performance index.

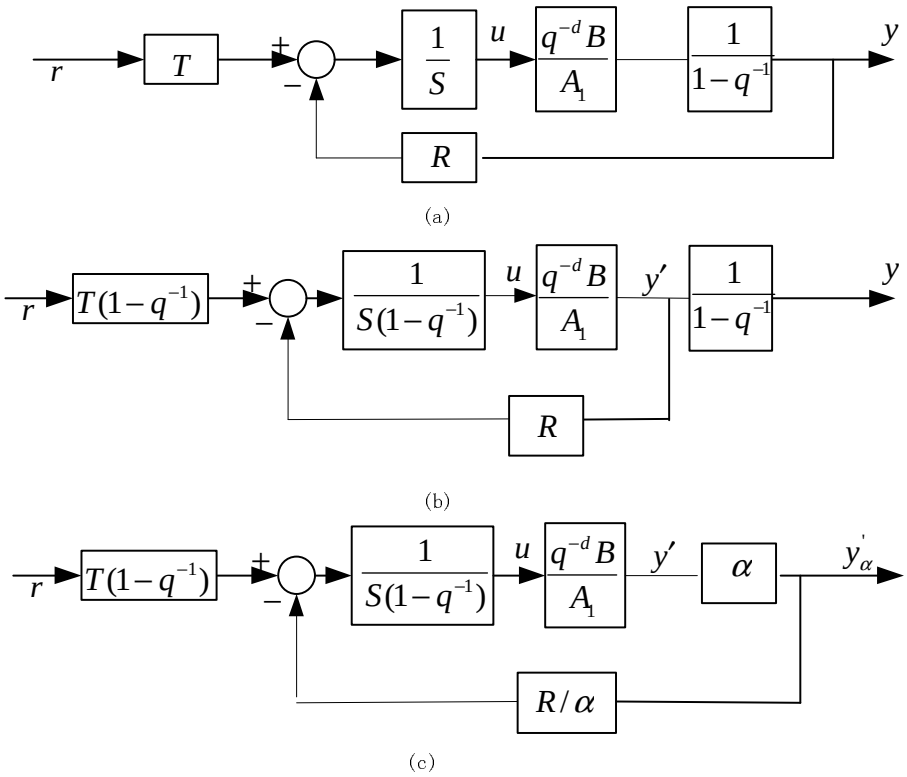

Figure A.3: Pre-processing for integrator and scaling in closed loop identification.

This test is based on the fact that the uncorrelation between the observation $\Phi(k)$ and the closed loop predictor leads to unbiased parameter estimates.

The cross-correlation is defined as

$$R(i) = \frac{1}{N}\sum_{i=1}^{N} \varepsilon_{CL}(k)\hat{y}(k-i) \tag{A-1}$$

The normalized cross-correlation is defined as

$$\begin{aligned} RN(i) &= \frac{R(i)}{\left[\left(\frac{1}{N}\sum_{i=1}^{N}\hat{y}^2(k)\right)\left(\frac{1}{N}\sum_{i=1}^{N}\varepsilon_{CL}^2(k)\right)\right]^{1/2}} \\ & i = 0, 1, 2, \ldots, \max(n_A, n_B + d) \end{aligned} \tag{A-2}$$

A good model should satisfy the validation criterion

$$|RN(i)| \leq \frac{\beta}{\sqrt{N}}, \quad i = 1, 2, \ldots, i_{\max} \tag{A-3}$$

where β is the confidence interval. A typical value of $\beta = 2.17$ corresponds to 97% level of confidence.

A.6 Experimental

The closed-loop identification is now used to obtain the model for the X and the Y axis. As the PRBS signals are basically a series of step inputs which are directly applied to the reference input, there is no trapezoid velocity profile for the axial drive to transit smoothly between different reference positions. Therefore it is wise to make the magnitude of the PRBS signals small to avoid the wear and tear to the mechanical transmission components, such as ball-screws and nuts. We have chosen the magnitude of PRBS to be 0.1 mm. The controller was chosen to be a proportional

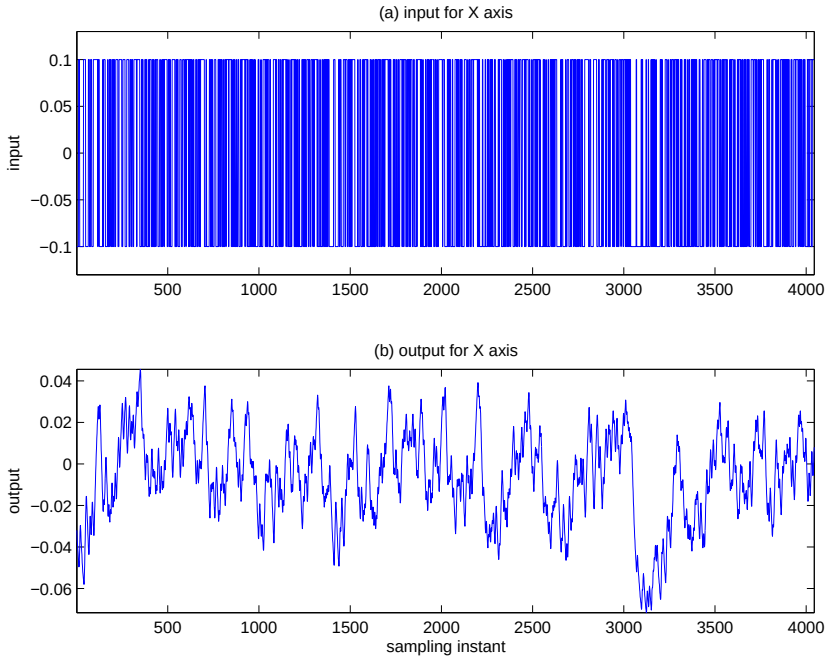

Figure A.4: Input and output for X axis: (a) PRBS reference input, (b)position output.

type controller. In this project, R, S and T are all set to be one. With the sampling frequency of 1 kHz, the input-output data is collected.

We take the X axis as an example. The PRBS reference inputs and the corresponding outputs for the X axis is shown in Fig. A.4. As the open-loop drive contains an integrator, we identify the relationship between the input, $u(k)$, and the variation of the output, $y(k) - y(k-1)$ as discussed in the previous section. Scaling was also carried out to make the input, $u(k)$, and the variation of the output, $y(k) - y(k-1)$, in the same range. The original and scaled variations in the outputs are shown in parts (a) and (b) of Fig. A.5 respectively.

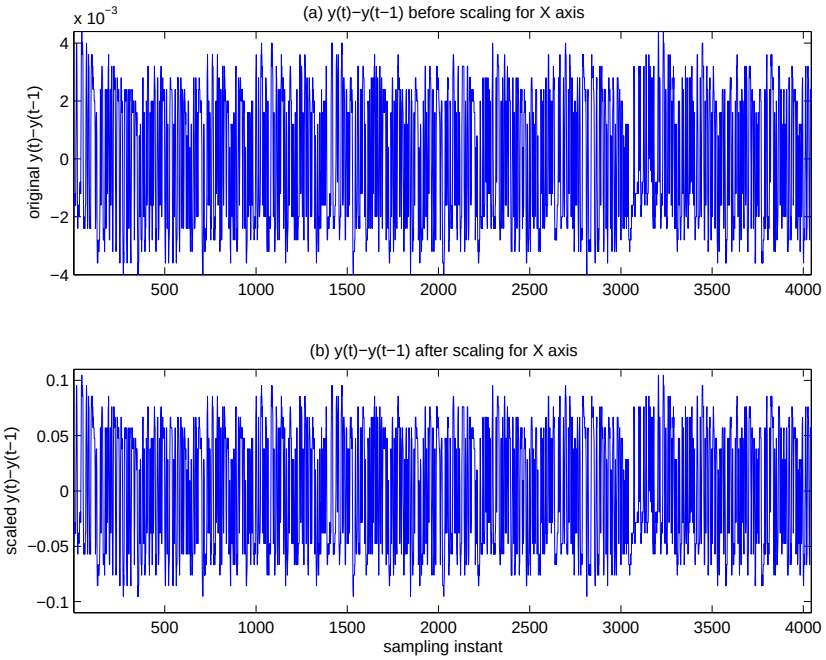

Figure A.5: Variation of the output $y(k) - y(k-1)$: (a) before scaling, (b) after scaling.

A.6.1 time delay d

The time delay d is unknown beforehand. We have tried different values from 1 to 3, and it was found when $d = 3$, the normalized cross-correlation gives the minimum value. Finally the time delay is chosen to be $d = 3$ throughout this project.

A.6.2 choose closed-loop identification method

As discussed before, there are four types of closed-loop identification, i.e., CLOE, A-CLOE, AF-CLOE and X-CLOE. However, there is no single method that works well for all cases. Statistical performance index can be in facilitating choose an appropriate closed-loop identification method for this particular application.

We therefore further check the uncorrelation performance, with the results shown in

Table A.1: Closed-loop error $R(0)$ and normalized cross-correlations for the X and Y axis

	CLOE	F-CLOE	AF-CLOE	X-CLOE	
X axis	$R(0)$	1.4325e-005	1.2356e-005	5.9695e-006	5.2395e-005
	$RN(\max)$	0.1298	0.1104	0.0488	0.3439
Y axis	$R(0)$	6.9970e-006	5.8240e-006	3.8311e-006	6.2955e-005
	$RN(\max)$	0.0662	0.0545	0.0337	0.4062

Table A.2: Coefficients for the X and Y model, for $d = 3$

$\frac{y(k)}{u(k)} = \frac{q^{-d}B(q^{-1})}{A(q^{-1})}$	q^0	q^{-1}	q^{-2}	q^{-3}
$B_x(q^{-1})$	0.00708911	0.00338940	0.00115112	0.0
$A_x(q^{-1})$	1.0	-1.79755510	1.05122750	-0.25367240
$B_y(q^{-1})$	0.00790339	0.00560520	0.00014294	0.0
$A_y(q^{-1})$	1.0	-1.52942945	0.64281390	-0.11338445

Table A.1. It is noted that AF-CLOE has lower correlation coefficients compared with CLOE, indicating less biased parameter estimation. Through these comparison, we finally choose AF-CLOE as the closed-loop identification method.

From this table, we can see that AF-CLOE gives the best results in terms of uncorrelations.

A.6.3 Coefficients for the model

After using the AF-CLOE method, and applying the post-processing the model according to Eq. A-2, the coefficients are listed in the Table A.2.

Appendix B

Author's Publications

Journal Papers

1. Xi, X.-C., Poo, A.-N., Hong, G.-S., Improving contouring accuracy by tuning gains for a bi-axial CNC machine, International Journals of Machines Tools and Manufacture, 49 (2009), 395-406.

2. Xi, X.-C., Poo, A.-N., Hong, G.-S. Improving CNC Contouring Accuracy by Integral Sliding Mode control, submitted to Mechatronics, 2008.

3. Xi, X.-C., Poo, A.-N., Hong, G.-S., Reducing static friction induced contour errors for a bi-axial CNC machine, submitted to International Journals of Machines Tools and Manufacture, 2008.

4. Xi, X.-C., Poo, A.-N., Hong, G.-S. Taylor Series Expansion Error Compensation for a Bi-axial CNC Machine, submitted to Journal of Dynamic Systems, Measurement and Control, 2008.

Conference Papers

1. Xi, X.-C., Poo, A.-N., Chou, S.-K., Factors affecting contour errors in cnc systems. In: Proceedings of 2005 9th International conference on mechatronics technology. Kuala Lumpur, Maylasia. ICMT-117, 2005.

2. Xi, X.-C., Poo, A.-N., Hong, G.-S. Taylor Series Expansion Error Compensation for a Bi-axial CNC Machine , 2008 IEEE International Conference on Systems, Man, and Cybernetics, Singapore, 12 - 15 October 2008.